

**UNIVERSIDAD AUTONOMA DE MADRID**

**ESCUELA POLITECNICA SUPERIOR**



**Grado en Ingeniería Informática**

# **TRABAJO FIN DE GRADO**

**Red Neuronal Bicuadrática Aplicada a la Igualdad de Género**

**Hugo García Calderón**  
**Tutor: David Domínguez Carreta**

**Mayo 2017**



# **Red Neuronal Bicuadrática Aplicada a la Igualdad de Género**

**AUTOR: Hugo García Calderón**  
**TUTOR: David Domínguez Carreta**

**Dpto. de Ingeniería Informática**  
**Escuela Politécnica Superior**  
**Universidad Autónoma de Madrid**  
**Mayo de 2017**



## Resumen (castellano)

Nuestro mundo está completamente informatizado y en constante evolución y esto es en parte a que el ser humano siempre intenta hacerse la vida más fácil. Ejemplo de esto son los teléfonos móviles, televisores inteligentes o los sistemas de navegación con los que ahora podemos hacer la compra, controlar nuestra casa inteligente o llegar a un destino que desconocemos sin siquiera saber dónde está, cosa que hace 30 años nos parecía imposible incluso pensar.

Gracias a esta digitalización e informatización de nuestra sociedad se nos abre un abanico muy grande de posibilidades para intentar hacer de este un mundo mejor, como por ejemplo, evitar la desigualdad de género. En la actualidad las mujeres de todo el mundo no cuentan con los mismos derechos, los mismos beneficios ni son tratadas con el mismo respeto que los hombres, es decir, sufren una desigualdad negativa que afecta a muchos ámbitos de su vida en su día a día.

Intentar reducir esta desigualdad está en manos de todo el mundo. Una de las acciones que podemos llevar a cabo es la concienciación de esta desigualdad de género para que las personas no contribuyan a aumentarla. Otra, por ejemplo, es dar más facilidades para los estudios de esta desigualdad para que podamos ver la evolución en el tiempo.

El objetivo de este Trabajo de Fin de Grado es el de aplicar inteligencia artificial para contribuir a los estudios de la igualdad de género. En concreto, dos algoritmos de redes neuronales son aplicados para ver cuál podría convenir usar para poder llevar a cabo este estudio. La red neuronal a utilizar es una red de tipo Hopfield.

## Palabras clave (castellano)

Red neuronal, Neurona, Memoria autoasociativa, Patrón, Igualdad de Género, Hopfield

## **Abstract (English)**

Our world is completely computerized and in a constant development as we as human beings always try to find ways to facilitate our life. Take for example the invention of mobile phones, smart TVs or navigation systems. These devices help us on our shopping trips, enable us to control our smart house or show us how to reach a yet unknown place. Thirty years ago, this would be something impossible to think of.

But this ongoing global digitalization also opens up a lot of possibilities which enable us to make this world a better place. For instance, improving the gender inequality. Up to the present-day women from all over the world are neither able to enjoy the same benefits nor entitled to the same rights as men. They are not treated with the same respect as men and consequently suffer from inequality that affects many areas in their daily life.

Every single one of us is capable of at least trying to reduce this inequality. One possible action would be to enhance the awareness of gender inequality so that people don't contribute unintentionally to its increase. Another option would be to provide tools for the inequality studies.

Therefore, the goal of this Bachelor thesis is to apply artificial intelligence to contribute to the studies of gender inequality. More specifically, within this thesis two neural network algorithms are applied in order to see which one would be better suitable to accomplish this goal. The neural network which is going to be used is the Hopfield network.

## **Keywords (inglés)**

Neural network, Neuron, Self-associative memory, Pattern, Gender Equality, Hopfield

## *Agradecimientos*

En general, a todas las personas que se han cruzado en mi vida, puesto que cada una de ellas me ha enseñado un poquito de la vida.

A mi tutor, David, por toda la ayuda prestada en los dos últimos años y por haber sabido aguantar lo pesado que puedo llegar a ser incluso por teléfono.

A mi segunda familia alemana en la que incluyo todos los amigos y compañeros de trabajo que estamos alejados de nuestro hogar por hacer la vida en el extranjero, y sobre todo, el día a día, maravilloso.

A mi actual jefe, Arcadio, por la motivación que me transmite día a día y por enseñarme que lo primero, antes que el trabajo, son las personas.

También quiero agradecer a Nicole, mi alemana favorita, por apoyarme y ayudarme a tomar decisiones tanto en mi vida personal como profesional y porque no importa lo que ocurra, ella siempre tiene unas palabras geniales para motivarte y alegrarte.

Por último quería agradecer a mi familia el gran sacrificio que han tenido que hacer a lo largo de los años para estar donde estoy, y porque si ahora soy como soy, es gracias a ellos. A mi hermano Daniel, mi hermana Nerea y mi madre Milagros, y por supuesto a mi padre Manolo, el cual me brindó una educación excelente y pude disfrutar de su compañía como padre y apoyo durante 21 años.

# INDICE DE CONTENIDOS

1	Introducción.....	12
1.1	Motivación.....	12
1.2	Objetivos.....	12
1.3	Organización de la memoria.....	12
2	Estado del arte .....	14
2.1	Igualdad de género.....	14
2.1.1	Introducción.....	14
2.1.2	Tipos .....	14
2.1.2.1	Trabajo.....	14
2.1.2.2	Educación .....	15
2.1.2.3	Tiempo.....	16
2.1.2.4	Violencia de género .....	17
2.1.3	Historia .....	18
2.1.4	Conclusión .....	18
2.2	Redes Neuronales .....	19
2.2.1	Introducción.....	19
2.2.2	Red Hopfield .....	20
2.2.2.1	Introducción.....	20
2.2.2.2	Funcionamiento .....	21
3	Diseño.....	25
3.1	Diseño y estructura del sistema .....	25
3.1.1	Estructura.....	25
3.1.2	Diseño .....	27
4	Desarrollo .....	29
4.1	Preproceso .....	29
4.2	Fase de aprendizaje.....	30
4.3	Fase de recuperación .....	31
4.3.1	Cálculo nuevo estado neuronal.....	31
4.3.2	Cálculo combinación lineal y función activación.....	32
4.4	Estudio del éxito de la recuperación.....	34
4.5	Elaboración de gráficas para la comparación .....	36
5	Integración, pruebas y resultados .....	37
5.1	Introducción.....	37
5.2	Resultados.....	39
6	Conclusiones y trabajo futuro.....	45
6.1	Conclusiones.....	45
6.2	Trabajo futuro .....	45
	Referencias .....	47
	Glosario .....	47
	Anexos.....	48
A	Gráficas de la desigualdad de género .....	48
B	Excepciones en el preproceso de los datos desde el fichero excel .....	51
C	¿Qué son los PPS? .....	52



# **INDICE DE FIGURAS**

ILUSTRACIÓN 2-1 INGRESO NETO MEDIO EQUIVALENTE (PPS) POR SEXO EN LA UE.....	15
ILUSTRACIÓN 2-2 PORCENTAJE DE MIEMBROS DE LOS CONSEJOS DE ADMINISTRACIÓN DE LAS EMPRESAS MÁS COTIZADAS POR SEXO EN LA UE .....	15
ILUSTRACIÓN 2-3 POBLACIÓN (15-74) QUE HA ALCANZADO LOS NIVELES 5 Y 6 DEL ISCED POR SEXO EN LA UE .....	16
ILUSTRACIÓN 2-4 HORAS MEDIAS DE TRABAJO DOMÉSTICO. DIFERENCIAS POR GÉNERO Y AÑO....	16
ILUSTRACIÓN 2-5 HORAS MEDIAS DE TRABAJO DOMÉSTICO DIFERENCIADAS POR LA SITUACIÓN LABORAL DE LAS DOS PARTES .....	17
ILUSTRACIÓN 2-6 PORCENTAJE DE MUJERES QUE HAN SUFRIDO VIOLENCIA FÍSICA POR SU PAREJA EN 2012.....	17
ILUSTRACIÓN 2-7 BRECHA GLOBAL GÉNERO 1 .....	18
ILUSTRACIÓN 2-8 BRECHA GLOBAL GÉNERO 2 .....	18
ILUSTRACIÓN 2-9 PARTES DE UNA NEURONA .....	19
ILUSTRACIÓN 2-10 MODELO DE LA RED HOPFIELD .....	20
ILUSTRACIÓN 3-1 FLUJO DEL PROGRAMA .....	26
ILUSTRACIÓN 4-1 CÓDIGO MATRIZ PESOS .....	31
ILUSTRACIÓN 4-2 CÓDIGO PORCENTAJE DE ERROR AL CREAR EL NUEVO ESTADO NEURONAL.....	32
ILUSTRACIÓN 4-3 CÓDIGO CREACIÓN DEL NUEVO ESTADO NEURONAL .....	32
ILUSTRACIÓN 4-4 CÓDIGO COMBINACIÓN LINEAL.....	32
ILUSTRACIÓN 4-5 CÓDIGO FUNCIÓN ACTIVACIÓN Y FUNCIONES SIGNO, ESCALÓN, MÓDULO Y COMBINADA.....	33
ILUSTRACIÓN 4-6 CÓDIGO ALCANZAR PUNTO DE ESTABILIDAD .....	33
ILUSTRACIÓN 4-7 CÓDIGO SOLAPAMIENTO 'M' Y 'N' .....	34
ILUSTRACIÓN 4-8 CÓDIGO CÁLCULO DE 'A'.....	34
ILUSTRACIÓN 4-9 CÓDIGO CÁLCULO DE 'Q'.....	35
ILUSTRACIÓN 4-10 CÓDIGO CÁLCULO DE 'L' .....	35
ILUSTRACIÓN 4-11 CÓDIGO GENERACIÓN GRÁFICAS.....	36

ILUSTRACIÓN 5-1 FALTA DE MEMORIA EN ECLIPSE .....	37
ILUSTRACIÓN 5-2 MEMORIA USADA POR ECLIPSE EN LAS EJECUCIONES Y POR LAS GRÁFICAS GENERADAS .....	38
ILUSTRACIÓN 5-3 SOLAPAMIENTO PARA PATRONES ALEATORIOS CON $C=0.5$ , NÚMERO DE ITERACIONES=500, 25% DE RUIDO, TIEMPO DE EJECUCIÓN TOTAL 5 MINUTOS Y $D=1$ .....	39
ILUSTRACIÓN 5-4 SOLAPAMIENTO PARA PATRONES ALEATORIOS CON $C=1$ , NÚMERO DE ITERACIONES=500, 25% DE RUIDO, TIEMPO DE EJECUCIÓN TOTAL 2 MINUTOS Y 20 SEGUNDOS Y $D=1$ .....	39
ILUSTRACIÓN 5-5 SOLAPAMIENTO PARA PATRONES ALEATORIOS CON $C=0$ , NÚMERO DE ITERACIONES=500, 25% DE RUIDO, TIEMPO DE EJECUCIÓN TOTAL 3 MINUTOS Y $D=1$ .....	40
ILUSTRACIÓN 5-6 SOLAPAMIENTOS PARA PATRONES REALES CON $C=0.5$ , NUMERO DE ITERACIONES =1000, 25% DE RUIDO, TIEMPO EJECUCIÓN TOTAL 5 SEGUNDOS Y $D=1$ .....	41
ILUSTRACIÓN 5-7 SOLAPAMIENTO PARA PATRONES REALES CON $C=1$ , NUMERO DE ITERACIONES =1000, 25% DE RUIDO, TIEMPO DE EJECUCIÓN TOTAL 5 SEGUNDOS Y $D=1$ .....	41
ILUSTRACIÓN 5-8 SOLAPAMIENTO PARA PATRONES REALES CON $C=0$ , NÚMERO DE ITERACIONES=1000, 25% DE RUIDO, TIEMPO DE EJECUCIÓN TOTAL 5 SEGUNDOS Y $D=1$ .....	42
ILUSTRACIÓN 5-9 SOLAPAMIENTO PARA PATRONES ALEATORIOS CON $C=0.5$ , NÚMERO DE ITERACIONES=500 Y 25% DE RUIDO. TIEMPO DE EJECUCIÓN TOTAL 5 MINUTOS Y $D=10$ .....	43
ILUSTRACIÓN 5-10 SOLAPAMIENTO PARA PATRONES ALEATORIOS CON $C=0.5$ , NÚMERO DE ITERACIONES=500 Y 25% DE RUIDO. TIEMPO DE EJECUCIÓN TOTAL 5 MINUTOS Y $D=20$ .....	43
ILUSTRACIÓN 5-11 SOLAPAMIENTO PARA PATRONES ALEATORIOS CON $C=0.5$ , NÚMERO DE ITERACIONES=500 Y 25% DE RUIDO. TIEMPO DE EJECUCIÓN TOTAL 5 MINUTOS Y $D=40$ .....	44
ILUSTRACIÓN A-0-1 DURACIÓN DE LA VIDA LABORAL POR SEXO EN LA UE .....	48
ILUSTRACIÓN A-0-2 PORCENTAJE DE EMPLEO A JORNADA COMPLETA POR SEXO EN LA UE .....	48
ILUSTRACIÓN A-0-3 TRABAJADORES QUE SON CAPACES DE COGERSE UNA O DOS HORAS DURANTE EL HORARIO DE TRABAJO PARA ATENDER ASUNTOS PERSONALES O FAMILIARES POR SEXO EN LA UE.....	49
ILUSTRACIÓN A-0-4 MIEMBROS DEL BANCO CENTRAL POR SEXO EN LA UE .....	49
ILUSTRACIÓN A-0-5 TASAS DE EMPLEO DE LOS TRABAJADORES ENTRE 15-54 Y DE LOS TRABAJADORES ENTRE 55-64 POR SEXO EN LA UE .....	49
ILUSTRACIÓN A-0-6 GANANCIA MEDIA MENSUAL (PPS) POR SEXO EN LA UE.....	50
ILUSTRACIÓN A-0-7 PARTICIPACIÓN DE LOS TRABAJADORES EN EL CUIDADO Y LA EDUCACIÓN DE LOS NIÑOS Y/O NIETOS, TODOS LOS DÍAS DURANTE UNA HORA O MÁS, POR SEXO EN LA UE..	50
ILUSTRACIÓN A-0-8 PARTICIPACIÓN DE LOS TRABAJADORES EN LA COCINA Y LAS TAREAS DOMÉSTICAS, TODOS LOS DÍAS DURANTE UNA HORA O MÁS, POR SEXO EN LA UE.....	50

ILUSTRACIÓN A-0-9 PORCENTAJE DE MUJERES QUE HAN SUFRIDO VIOLENCIA SEXUAL POR SU PAREJA EN 2012 .....	51
--	----

## INDICE DE TABLAS

<u>TABLA 3-1 EJEMPLO DE FORMATO DEL FICHERO EXCEL</u> .....	26
<u>TABLA 4-2 FORMATO DEL FICHERO CREADO PARA LA FÁCIL COMPARACIÓN DE PATRONES</u> .....	30

## INDICE DE ECUACIONES

ECUACIÓN 2-1 CÁLCULO DE MATRIZ DE PESOS BILINEAL 'W' .....	21
ECUACIÓN 2-2 CÁLCULO MATRIZ DE PESOS BICUADRÁTICA 'V' .....	21
ECUACIÓN 2-3 CREACIÓN DEL NUEVO ESTADO NEURONAL CON RUIDO.....	22
ECUACIÓN 2-4 CÁLCULO COMBINACIÓN LINEAL SINAPSIS BILINEAL 'H' .....	22
ECUACIÓN 2-5 CÁLCULO COMBINACIÓN LINEAL SINAPSIS BICUADRÁTICA 'G' .....	22
ECUACIÓN 2-6 CÁLCULO CÁLCULO FUNCIÓN ACTIVACIÓN COMBINADA DE RED BILINEAL Y BIMODAL.....	23
ECUACIÓN 2-7 CÁLCULO FUNCIÓN ACTIVACIÓN RED BILINEAL.....	23
ECUACIÓN 2-8 CÁLCULO FUNCIÓN ACTIVACIÓN RED BIMODAL .....	23
ECUACIÓN 2-9 CÁLCULO SOLAPAMIENTO BILINEAL 'M' .....	23
ECUACIÓN 2-10 CÁLCULO SOLAPAMIENTO BICUADRÁTICO 'L' .....	24
ECUACIÓN 2-11 CÁLCULO NÚMERO DE ACTIVOS PARA LA RED BICUADRÁTICA 'A' .....	24
ECUACIÓN 2-12 CÁLCULO SOLAPAMIENTO-ACTIVIDAD BICUADRÁTICO 'N' .....	24
ECUACIÓN 2-13 CÁLCULO ACTIVIDAD NEURONAL 'Q' .....	24

# 1 Introducción

---

## 1.1 Motivación

La motivación de este proyecto es la de olvidarnos por un momento de todo lo que hemos aprendido y estudiado desde el punto de vista técnico y centrarnos en mejorar el mundo. ¿Cómo lo hacemos? Usando todo lo que hemos aprendido y estudiado desde el punto de vista técnico.

Hace tiempo la tecnología y el arte y los temas sociales estaban muy alejados unos de otros, pero poco a poco se han ido necesitando uno al otro. Por ejemplo, gracias a la tecnología somos capaces de hacer tours en realidad virtual en los que podemos visitar museos o ver los monumentos más sorprendentes.

Es gracias a este acercamiento por lo que se nos da la oportunidad de aportar y trabajar en algo en lo que, siguiendo nuestro pasado informático, no haríamos, como es un estudio de la desigualdad de género.

## 1.2 Objetivos

La igualdad de género es un tema muy crítico últimamente en el que se están haciendo varios estudios para saber si la brecha entre hombres y mujeres crece o decrece.

Se parte de la base (aunque más adelante se verá que no siempre es así) que a medida que pasan los años la brecha decrece, por lo tanto, usando la red neuronal que se va a usar se comprobará (a nivel mundial) si esta brecha crece o decrece de una manera rápida si tenemos los datos necesarios para un determinado año. En tal caso veremos que la brecha decrece si una nueva entrada la reconoce como un patrón cercano o en el tiempo o veremos si la brecha crece en caso de que el patrón que se reconozca sea más antiguo.

El objetivo principal es comprobar si la red neuronal ternaria es más efectiva que la binaria para este específico problema. Para ello se emplea un modelo de aprendizaje bicuadrático en una red hopfield de tipo hebianna.

En anteriores trabajos se ha demostrado que usar 3 estados y la red ternaria en vez de la red binaria mejora la recuperación. Esta mejora de recuperación depende mucho de los datos de entrada con los que se trabajan, así que esta vez puede ser distinto.

## 1.3 Organización de la memoria

La primera sección de este documento es una introducción al trabajo e incluye una descripción de las motivaciones y los objetivos del estudio.

El segundo capítulo recoge el análisis del estado del arte sobre la igualdad de género, en él se habla de qué es la igualdad de género y se intenta concienciar de la desigualdad existente actualmente. Además, hay una introducción a las redes neuronales y al uso de la memoria asociativa así como una explicación detallada teórica del algoritmo a utilizar.

En la tercera sección se describe la estructura y el diseño a alto nivel del programa, es decir, qué partes tiene y qué hace cada parte.

Es en la cuarta sección donde se explica cómo se ha desarrollado todo lo explicado en la tercera sección. En esta sección se entra en detalles técnicos que están acompañados de algunas fotos con pseudocódigo para ayudar el entendimiento del algoritmo.

En la quinta sección se ven las distintas pruebas a las que han sido sometidas el programa y los distintos resultados de estas pruebas. También en esta sección se explica el comportamiento de estos resultados.

En el sexto capítulo se detallan las conclusiones y posibles líneas de trabajo futuras.

Todo material adicional que no sea de vital importancia para entender y seguir el documento se encuentra en la sección de anexos.

## 2 Estado del arte

---

### 2.1 Igualdad de género

#### 2.1.1 Introducción

Dado que el estudio se basa en construir un tipo de red neuronal para datos de igualdad de género, primero hay que saber **qué es la igualdad de género**.

La igualdad de género es un tema que preocupa mucho en la actualidad y es por eso que todo el mundo ha escuchado hablar de ella, ya sea en la radio, noticias, amigos o internet, pero haber escuchado el tema no significa exactamente saber qué es y qué engloba. Se puede comprobar esto mismo si se observan los siguientes ejemplos de respuestas de personas preguntadas en México D.F. sobre qué es la igualdad de género [1] :

- Hombres y mujeres somos iguales
- Personas del mismo sexo
- Mismas necesidades y mismos derechos
- Oportunidades iguales en todos los sentidos (trabajo, educación, desarrollo personal psicológico, cultural y artístico)
- Tenemos mismos derechos hombres y mujeres
- Mismos derechos, responsabilidades y obligaciones
- No entiendo nada de eso

Por lo tanto, **la igualdad de género** implica que hombres y mujeres deben recibir los **mismos beneficios, las mismas sentencias y ser tratados con el mismo respeto** [2]. Esta definición engloba muchos aspectos que se tratan en el siguiente punto.

#### 2.1.2 Tipos

Cuando se habla de que no hay igualdad de género, esto puede englobar distintos ámbitos como:

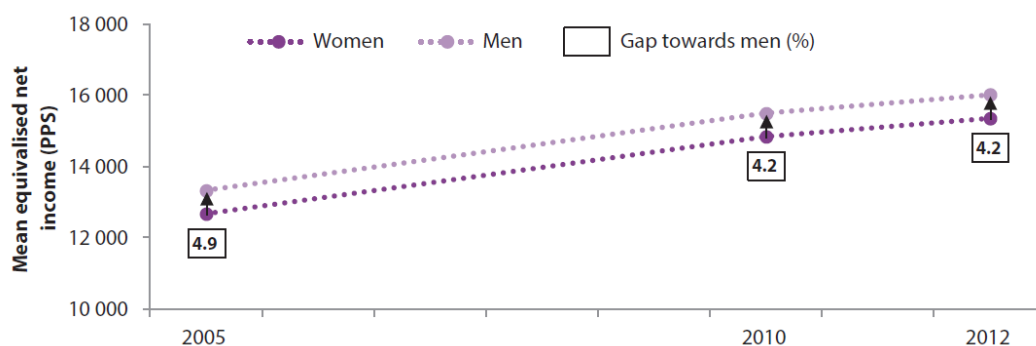
##### 2.1.2.1 Trabajo

Dentro del trabajo pueden ocurrir muchas desigualdades entre hombres y mujeres.

El porcentaje de empleo y/o paro, el salario medio, la permisividad de los jefes a sus trabajadores o el porcentaje de altos cargos en las empresas son solo algunos ejemplos de las desigualdades que se pueden sufrir en el trabajo. [3]

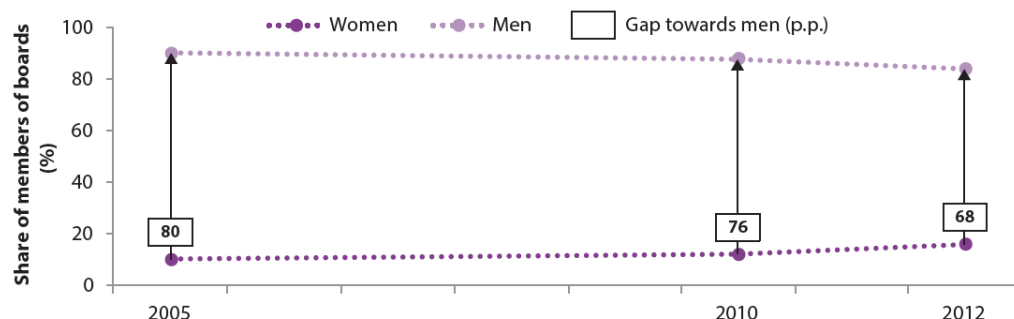
En promedio, las mujeres siguen ganando en todo el mundo un 24 por ciento menos que los hombres, además, si se eliminasen las desigualdades entre hombres y mujeres en el ámbito laboral se añadiría un 26 por ciento más de riqueza al Producto Interior Bruto (PIB) mundial y contribuiría al crecimiento de la economía pública y privada. [4]

En las siguientes gráficas se puede ver esta desigualdad de género en el trabajo.



Source: Eurostat, EU-SILC (ilc\_di03).

**Ilustración 2-1 Ingreso neto medio equivalente (PPS) por sexo en la UE [3]**



Source: EC-DG Justice, WMID.

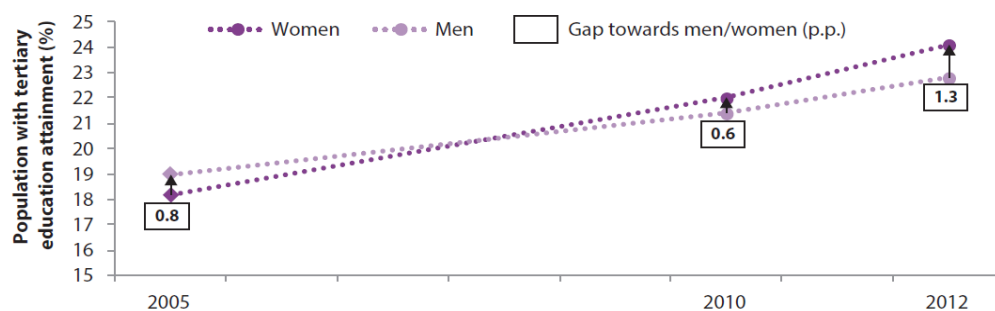
**Ilustración 2-2 Porcentaje de miembros de los consejos de administración de las empresas más cotizadas por sexo en la UE [3]**

### 2.1.2.2 Educación

Que haya igualdad de género en la educación no es sólo un derecho humano básico, sino que además es crucial para acelerar el desarrollo sostenible. Ha sido demostrado una y otra vez que empoderar a las mujeres y niñas tiene un efecto multiplicador y ayuda a promover el crecimiento económico y el desarrollo a nivel mundial. [5]

Hasta el día de hoy ha habido una evolución en cuanto a la lucha contra la discriminación de género, como por ejemplo, la introducción de un sistema educativo mixto con la Ley General de Educación en 1970. No obstante, aún queda un largo camino por recorrer para alcanzar el objetivo de tener un modelo **coeducativo** en el que se aspira a un reparto equitativo, a la cooperación, el compañerismo activo y respeto mutuo. [6]

Como se ve en la siguiente imagen, ha habido una inclusión de la mujer dentro de la educación hasta el punto de que ahora hay más mujeres que hombres que alcanzan los niveles 5 y 6 del ISCED (ciclo formativo de grado superior y grado universitario respectivamente)

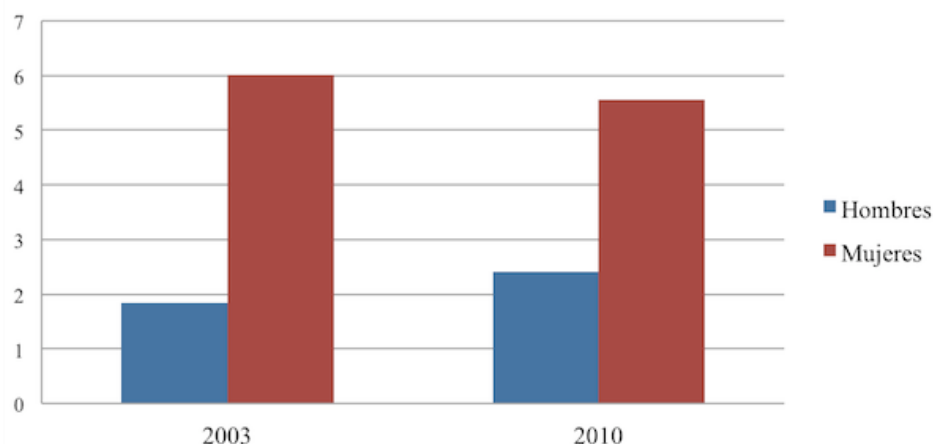


Source: Eurostat, SES (earn\_ses10\_20).  
Note: EU-28 average was not available in 2005, EU-27 average was used.

**Ilustración 2-3 Población (15-74) que ha alcanzado los niveles 5 y 6 del ISCED por sexo en la UE [3]**

### 2.1.2.3 Tiempo

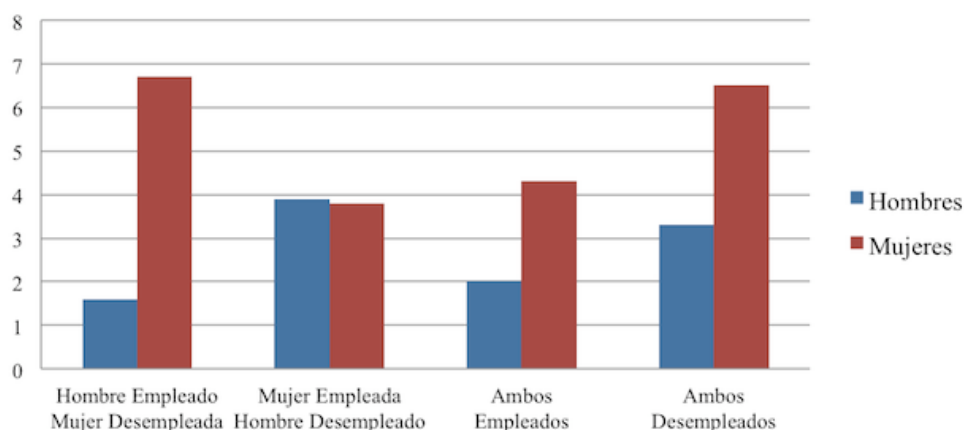
En la siguiente gráfica se puede observar una clara desigualdad en cuanto al tiempo que dedican hombres y mujeres a las tareas del hogar o el cuidado y educación de los niños. Se puede observar que ha habido una mejoría y una tendencia hacia la igualdad, pero que aún hay una clara desigualdad.



**Ilustración 2-4 Horas medias de trabajo doméstico. Diferencias por género y año. [7]**

Se podría pensar que esta desigualdad de género está motivada porque, aún en la época que vivimos, el hombre es el sustento económico familiar y la mujer se encarga de las tareas de la casa, pero el siguiente gráfico nos muestra que, aunque las parejas españolas de doble salario son las más igualitarias en lo global, las mujeres (de todos los perfiles laborales) emplean mucho más tiempo al trabajo doméstico del que esperaríamos en una sociedad equitativa.[7]





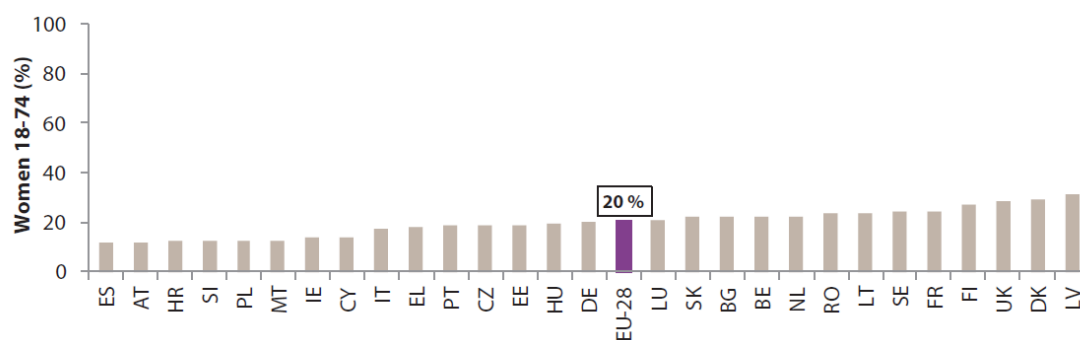
**Ilustración 2-5 Horas medias de trabajo doméstico diferenciadas por la situación laboral de las dos partes [7]**

### 2.1.2.4 Violencia de género

La violencia de género es el punto más importante y crítico, pues se habla de maltrato físico y psicológico que conduce incluso al asesinato. Sólo durante el año 2016 ha habido 56 asesinatos por violencia de género en España.

Se podría escribir otro trabajo de fin de grado sólo para la violencia de género, pero el tema a tratar es diferente así que sólo se menciona con algunos ejemplos.

Algunos datos alarmantes son que, de media, un 20% de mujeres han sufrido violencia física en la UE, muy por encima del 12% de España. Aún así, estos datos revelan que en 2012 más de 35 millones de mujeres en la UE han sido víctimas de violencia física



Source: FRA (2014a), EU-wide Survey on Violence against Women database (DVS\_A08).

**Ilustración 2-6 Porcentaje de mujeres que han sufrido violencia física por su pareja en 2012 [3]**

### 2.1.3 Historia

Como se ha descrito anteriormente, todo el mundo ha escuchado hablar sobre la igualdad de género, y quizá es por eso por lo que ha perdido importancia últimamente. Y es que el resultado del informe sobre la **Brecha de Género** a nivel Global en 2016 dice que la desigualdad económica entre hombres y mujeres se está haciendo mayor debido a la crisis de 2008, por lo que las perspectivas de tener completa igualdad de género en el ámbito laboral se han aplazado hasta el año 2186[8], mientras que en el 2015 la predicción fue de cerrar esta brecha en 2133. [9]

Esta Brecha Global de Género se presentó por primera vez en 2006 y se utiliza para entender en qué medida los recursos y oportunidades se están distribuyendo entre hombres y mujeres en los 142 países que se analizan. [10]

En particular, el índice analiza los puntos descritos en el anterior punto como son la participación económica y oportunidad (salarios, empleo capacitado) y la educación (acceso a niveles de educación), así como la partición política y la salud.

Este índice es de gran ayuda a la hora de poder hacer un estudio sobre cómo evoluciona la igualdad de género año a año.

### 2.1.4 Conclusión

La mayoría de estos datos y gráficos son pertenecientes a la Unión Europea y de ellos se puede sacar la conclusión de que la mujer no cuenta con los mismos beneficios, las mismas sentencias y no está tratada con el mismo respeto. Si es cierto que los resultados muestran un progreso entre los años 2005 y 2012. [3] En la parte de [Anexos](#) hay un apartado dedicado con más gráficas interesantes sobre la desigualdad de género.

Si queremos extrapolar estos datos de Europa a todo el mundo hay una clara desigualdad de género mucho mayor que la descrita anteriormente en la que sólo Estados Unidos es equiparable a Europa como se puede ver en las siguientes imágenes.

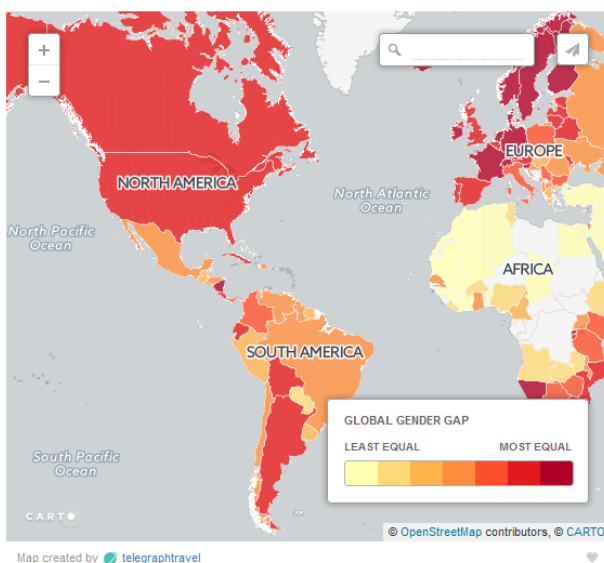


Ilustración 2-7 Brecha Global Género 1 [18]

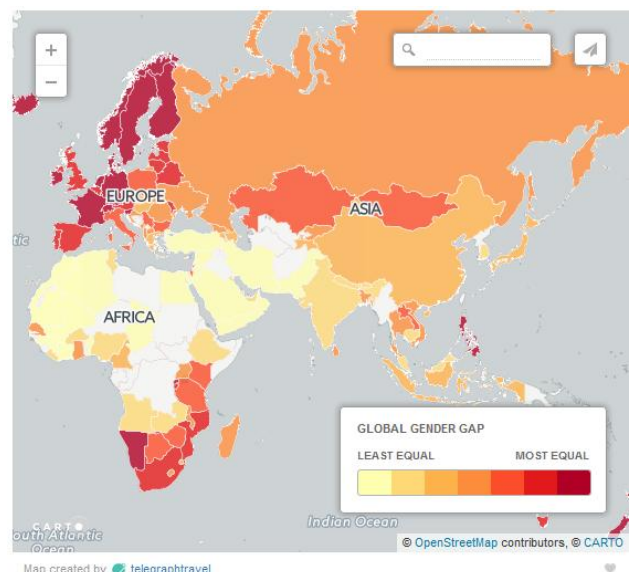


Ilustración 2-8 Brecha Global Género 2 [18]

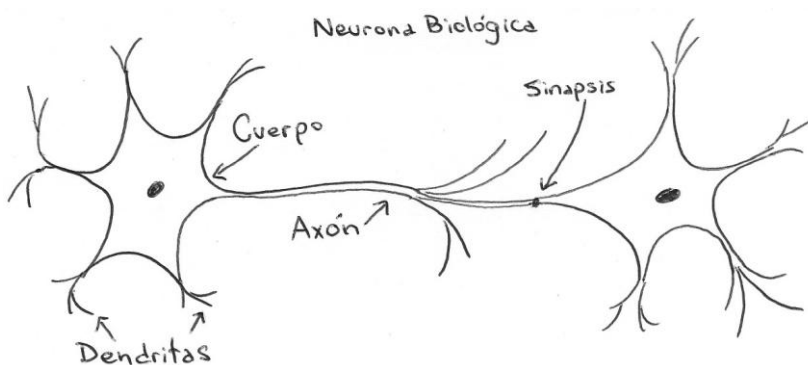
## 2.2 Redes Neuronales

### 2.2.1 Introducción

A lo largo de la historia el hombre siempre ha querido mejorar su calidad de vida y reducir los esfuerzos (físicos y mentales) a la hora de realizar tareas o resolver problemas.

Las redes neuronales están basadas en la inteligencia artificial, y esta es un intento por descubrir y describir aspectos de la inteligencia humana que pueden ser simulados mediante máquinas, así pues vemos que las redes neuronales es otra manera de facilitar un trabajo al ser humano, en este caso, es el de emular ciertas características propias de los humanos, como la capacidad de aprender. Hay muchos problemas en la vida que no son capaces de ser resueltos aplicando un algoritmo si no la experiencia, así pues las redes neuronales intentan emular esta experiencia para la resolución de esos problemas, para ello usan la célula fundamental del sistema nervioso humano, la neurona. [11]

**La neurona artificial** funciona de la misma manera que la biológica, en la que las dendritas serían las entradas de la red neuronal, el núcleo sería el valor de la neurona y la transferencia se lleva a cabo en la sinapsis, que es donde se localiza la inteligencia y donde se conectan las neuronas entre sí.



**Ilustración 2-9 Partes de una neurona [19]**

Estas entradas se procesan a través de la red neuronal con el objetivo de lograr una salida, la cual será correcta sólo si el aprendizaje de las entradas ha sido correcto. Para poder llegar a obtener una salida correcta hay que modificar esta fase de aprendizaje en la que se entrena a la red neuronal. En esta fase las entradas nunca pueden ser modificadas pero sí los pesos de cada una de las conexiones (más adelante se habla de los pesos).

Se pueden distinguir dos métodos principales de aprendizaje: el aprendizaje supervisado y el aprendizaje no supervisado.

**El aprendizaje supervisado** se caracteriza porque el proceso de aprendizaje se realiza de una manera controlada que determina la respuesta que debería producir una entrada específica. Si esta respuesta es errónea entonces se modificarán los pesos de las conexiones. Este algoritmo se repetirá hasta que la salida sea la deseada. [11]

En el **aprendizaje no supervisado** la red no recibe ninguna información sobre si la salida es correcta o no. En algunos casos (el estudio que se describe en este trabajo de fin de grado, por ejemplo) la salida contiene grado de familiaridad con la información de aprendizaje en la entrada. [11]

La ventaja principal de una red neuronal frente a otro algoritmo computacional es que pueden generalizar, es decir, frente a entradas desconocidas son capaces de dar salidas aproximadas a las deseadas.

## 2.2.2 Red Hopfield

### 2.2.2.1 Introducción

La red Hopfield es un tipo de red monocapa de aprendizaje no supervisado de tipo hebbiano en la que las neuronas tienen valores binarios (o ternarios para la nueva variante que se estudia en este trabajo de fin de grado) y están conectadas todas entre sí menos consigo misma.

En esta red la salida de cada neurona se calcula y retroalimenta como entrada, calculándose otra vez hasta llegar a un punto de estabilidad. Estos cambios de salidas a entradas cada vez van siendo más pequeños hasta llegar a un punto (teórico) en el que no hay cambio y se alcanza la estabilidad.

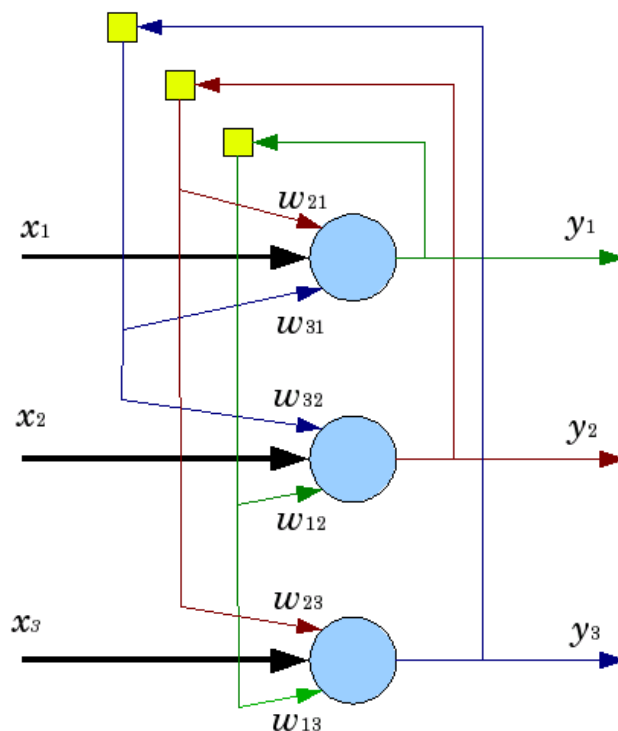


Ilustración 2-10 Modelo de la red Hopfield [20]

La finalidad de esta red es aprender (almacenar) un conjunto de patrones de forma que cuando se presente un nuevo patrón la red sepa reconocer y asignar este nuevo patrón a uno de los ya aprendidos.

Esta red es capaz de aprender y reconocer patrones porque es una red autoasociativa, es decir, recupera información a partir del conocimiento parcial del contenido y no basándose en la localización del almacenamiento. Un ejemplo es que no busca el dato en la posición de memoria 'x' sino que recuerda ciertas características de ese dato. [12]

Las aplicaciones de esta red son básicamente el reconocimiento de imágenes y voz puesto que almacenan patrones y luego responden con un patrón aprendido a un nuevo estímulo inicial.

El objetivo de este TFG, como ya hemos descrito, es la comparación de los resultados al usar una red Hopfield con 2 estados (binaria) y con 3 estados (ternaria). A continuación se describe el proceso de forma teórica para hallar esos resultados.

### **2.2.2.2 Funcionamiento**

#### **2.2.2.2.1 Fase de aprendizaje**

Como se ha descrito anteriormente, el modelo Hopfield está basado en una red monocapa en la cual las neuronas están conectadas todas entre sí menos consigo mismas, por lo tanto la estructura del aprendizaje es una matriz de pesos cuadrada, simétrica y donde la diagonal principal es 0.

Se calcula de la siguiente manera:

$$W_{ij} = \begin{cases} \sum_{\mu=1}^p X_i^{\mu} X_j^{\mu} & \text{si } i \neq j \\ 0 & \text{si } i = j \end{cases}$$

**Ecuación 2-1 Cálculo de Matriz de Pesos Bilineal 'W'**

$$V_{ij} = \begin{cases} \sum_{\mu=1}^p (X_i^{\mu} - a)^2 (X_j^{\mu} - a)^2 & \text{si } i \neq j \\ 0 & \text{si } i = j \end{cases}$$

**Ecuación 2-2 Cálculo Matriz de Pesos Bicuadrática 'V'**

#### 2.2.2.2.2 Fase de recuperación

Una vez ha terminado la fase de aprendizaje existe una fase de recuperación en la que se comprueba cómo de bien ha ido esa fase de aprendizaje. Para ello, lo primero que hay que hacer es crear un nuevo estado neuronal en base al patrón aprendido anteriormente de la siguiente forma:

$$Y^0 = X + R = \text{Patrón aprendido} + \text{ruido}$$

#### Ecuación 2-3 Creación del nuevo estado neuronal con ruido

Esta ecuación está expresada como una suma pero realmente lo que se hace es modificar ciertos valores (depende del porcentaje de ruido) dentro del patrón aprendido. En la parte de [Desarrollo](#) se explica con más detalle.

Una vez conseguido este nuevo estado neuronal hay que calcular la combinación lineal, es decir, el nuevo estado neuronal multiplicado por la matriz de pesos.

$$H_i = C * \sum_{j(D)}^n W_{ij} Y_j = \text{Matriz Pesos bilineal} * \text{Estado neuronal}$$

#### Ecuación 2-4 Cálculo Combinación lineal sinapsis bilineal 'H'

$$G_i = (1 - C) * \sum_{j(D)}^n V_{ij} Y_j^2 = \text{Matriz Pesos bicuadrática} * (\text{Estado neuronal})^2$$

#### Ecuación 2-5 Cálculo Combinación lineal sinapsis bicuadrática 'G'

El sumatorio de las ecuaciones anteriores tiene el punto inicial en una variable aleatoria módulo 'D' y la suma tiene incremento 'D'. De esta variable 'D' se habla en la parte de [Resultados](#) y sirve para "desconectar" neuronas de otras. De todas formas, esta parte se explica con más detalle en la sección de [Desarrollo](#). Por otra parte, se incluye una variable 'C' que es la que indica si la red neuronal es bilineal (C=1), bimodal (C=0) o si por el contrario una combinación de las dos, una red ternaria (cualquier valor de C distinto de 0 y 1).

Para hacer que esta recuperación sea óptima Hopfield establece estados que se van modificando como vamos a ver a continuación. En el momento en el que estas variaciones sean nulas (o después de 'x' variaciones) es cuando se llega a un punto local estable en el que se garantiza que la recuperación es óptima. Este paso se puede dividir en 3 partes:

1. Se calcula H o G para el instante t y se pasa por la función de activación (ver más adelante).
2. Se repite el paso número 1 para el instante t+1.
3. Se comparan los resultados de los dos primeros pasos. En caso de que sean iguales el proceso termina, en caso contrario se repiten los pasos hasta que los resultados sean iguales o transcurra un tiempo definido.

Como ya se ha mencionado a lo largo de este documento, cada una de las neuronas de la red define su salida en función de una o varias entradas que luego es transferida a las siguientes neuronas. Esta salida es proporcionada por la **función de activación** que, emulando a la neurona biológica, da un estado de excitación (1) o no excitación (-1). Por supuesto para este nuevo modelo de tres estados se añade otro más (0).

Hay varios tipos de función de activación, pero en concreto la red de la que se está haciendo el estudio fue ideada para trabajar con valores binarios (o ternarios en este caso), por lo que las funciones de activación de las neuronas son de tipo signo y escalón. Estas fórmulas son aplicadas a la combinación lineal calculada anteriormente:

$$Y^* = f1(H) f2(|H| + G)$$

**Ecuación 2-6 Cálculo función activación combinada de red bilineal y bimodal**

$$f1 = \text{signo}$$

**Ecuación 2-7 Cálculo Función activación red bilineal**

$$f2 = \text{escalón}$$

**Ecuación 2-8 Cálculo Función activación red bimodal**

La función signo es una función que obtiene el signo de cualquier número real que se tome por entrada. En este caso, es 1 si el número es mayor que 0 y -1 si es menor.

La función escalón es igual que la función signo con la peculiaridad del número 0. En este caso, cuando el número es 0, el valor de la función escalón también es 0. De esta manera se consiguen los 3 estados.

Lo interesante es ver cómo de correcta es la recuperación, y para ello hay que comparar el solapamiento bilineal (m) con el bicuadrático (l).

El solapamiento bilineal se calcula de la siguiente manera:

$$m = \frac{1}{a N} \sum_{i=1}^n X_i Y_i^*$$

**Ecuación 2-9 Cálculo Solapamiento bilineal 'm'**

Donde 'a' es el número de activos dentro de la neurona (en este caso, como los datos en la red bilineal son -1s y 1s, es el 100%) y 'N' es el número total de neuronas.

El valor de 'm' oscila entre [-1, 1].

Para calcular el solapamiento bicuadrático es necesario calcular el siguiente valor:

$$l = \frac{n - q}{1 - a}$$

**Ecuación 2-10 Cálculo Solapamiento bicuadrático 'l'**

El valor de 'l' oscila entre [-1, 1] y las demás variables se describen a continuación.

$$a = \frac{1}{N} \sum_{i=1}^n X_i^2$$

**Ecuación 2-11 Cálculo Número de activos para la red bicuadrática 'a'**

El valor de 'a' expresa el número de activos dentro de la neurona y en este caso no es 100% porque se cuenta con el valor 0 así que el valor que puede tomar 'a' oscila entre [0,1].

$$n = \frac{1}{a N} \sum_{i=1}^n X_i^2 Y_i^2$$

**Ecuación 2-12 Cálculo Solapamiento-actividad bicuadrático 'n'**

Por otra parte, 'n' es el número de neuronas que coinciden con un patrón en activo (1s ó -1s) y su valor oscila entre [-1, 1]:

$$q = \frac{1}{N} \sum_{i=1}^n Y_i^2$$

**Ecuación 2-13 Cálculo Actividad neuronal 'q'**

Por último, 'q' es la actividad media neuronal y su valor oscila entre [0, 1].

Todas estas fórmulas y el algoritmo de la red ternaria han sido extraídas y adaptadas según la solución necesaria del artículo "Three state neural network: from Mutual Information to the Hamiltonian" elaborado por David R. Domínguez Carreta.



## 3 Diseño

---

### 3.1 Diseño y estructura del sistema

En este capítulo se va a explicar el diseño a alto nivel después de los detalles teóricos explicados anteriormente.

Los patrones que se usarán como neuronas de entrada serán los correspondientes a la transformación del fichero excel de igualdad de género (muchas gracias a Mar de la facultad de económicas por el fichero con los datos). En la parte de [Desarrollo](#) se explica cómo se hace este preproceso tanto para los patrones binarios como para los ternarios.

Las implementaciones han sido realizadas en un lenguaje de alto nivel y orientado a objetos (Java) haciendo uso directamente de los datos desde el fichero excel. No se necesita crear una base de datos ni usar un lenguaje de programación con mejor rendimiento debido a que los cálculos son perfectamente soportables con un computador estándar (en este caso especial un ordenador portátil con Intel Core i5-4210H a 2.9 GHz y 12 GB de memoria RAM) en los que los tiempos de ejecución no son muy altos (en la parte de [Resultados](#) se pueden observar estos tiempos de ejecución)

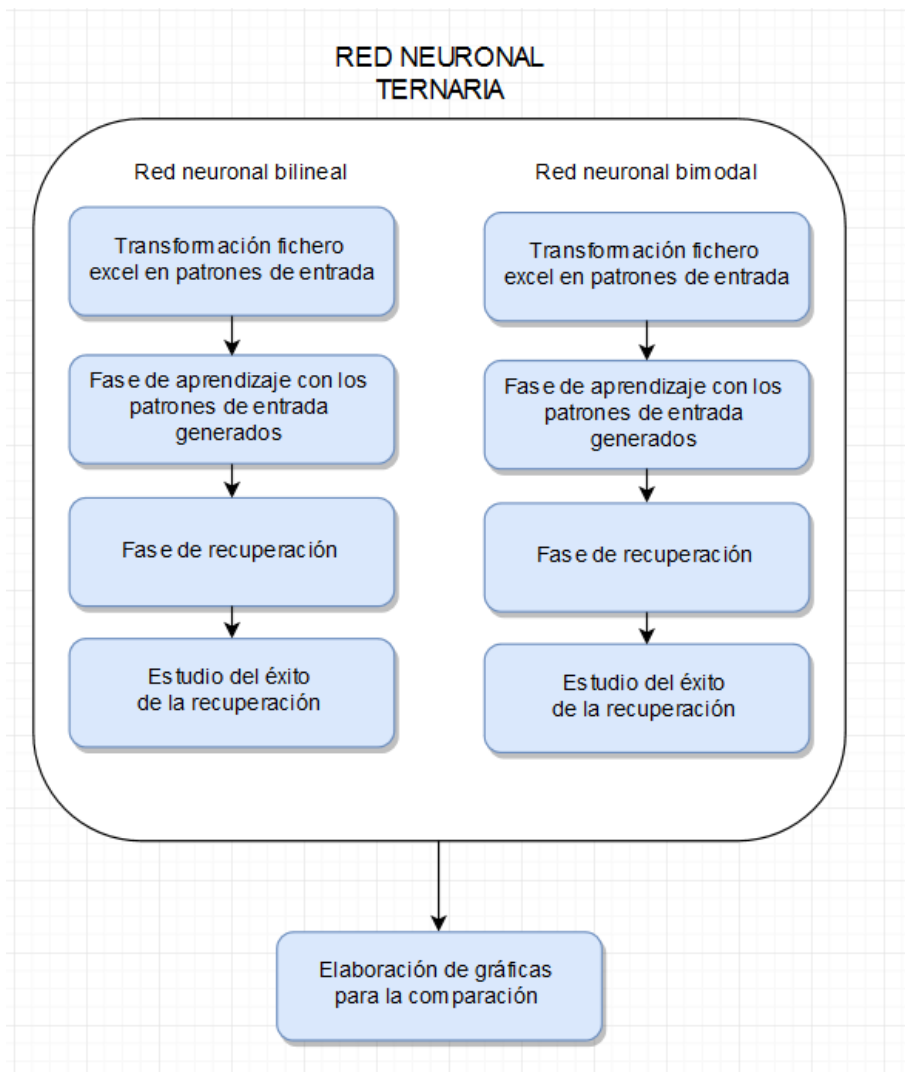
#### 3.1.1 Estructura

Como ya se ha descrito, el objetivo de este trabajo de fin de grado es la comparación del resultado de la recuperación de patrones usando una red binaria (1s y -1s) y usando una red ternaria (1s, 0s y -1s).

Como ya se ha visto anteriormente, el programa se ejecuta con un valor (de 0 a 1) en una constante C que es la que combina los 3 posibles estados. Este valor de C determina qué porcentaje de red bilineal (1s y -1s) y red bimodal (1s y 0s) se quiere usar para construir la red ternaria.

Por lo tanto, el programa repetirá los procesos realizados dos veces para obtener los resultados para ambas redes y así poder construir la red ternaria.

El flujo del programa es el siguiente:



**Ilustración 3-1 Flujo del programa**

- **Transformación fichero Excel en patrones entrada.** En el fichero de entrada excel están descritas algunas de las leyes (18 en total) a tomar en cuenta para saber si un país es machista o feminista. El fichero contiene la muestra desde 1960 hasta 2010 para un total de 100 países con la siguiente estructura

Year	Country	Ley 1	Ley 2	Ley 3	Ley 4	Ley 5	Ley 6
1960	Algeria	N	Y	N	Y	N	Y
..	Algeria	Y	Y	Y	Y	Y	Y
2010	Algeria	(cust)	N	(cust)	N	(cust)	N
1960	Zimbaue	Y	N	Y	N	Y	N
..	Zimbaue	N	Y	N	Y	N	Y
2010	Zimbaue	Y	N	Y	N	Y	N

**Tabla 3-1 Ejemplo de formato del fichero excel**

El objetivo es conseguir un fichero en el que hay una línea por patrón (año) y todas las leyes de todos los países con los valores 1, -1 y 0.

- **Fase de aprendizaje con los patrones de entrada generados.** Mediante los algoritmos descritos en la sección anterior para la red Hopfield se genera la matriz de Pesos para cada uno de los patrones de entrada.
- **Fase de recuperación.** Consiste en generar un nuevo patrón de entrada basado en uno de los aprendidos con un porcentaje de ruido para comprobar si nuestra red es capaz de recuperar este nuevo patrón y reconocerlo. Esta parte es la que consumirá más tiempo de ejecución, pues, como se ve en la sección de Desarrollo, hay que iterar 'x' veces por cada uno de los patrones hasta llegar a un estado estable.
- **Estudio del éxito de la recuperación.** Mediante los algoritmos descritos en la sección anterior se comprobará el suceso de esta recuperación calculando los solapamientos de ambas redes
- **Elaboración de gráficas para la comparación.** Gracias a los resultados obtenidos en la fase anterior se elabora una comparación donde se pueden observar los distintos valores para la red ternaria. Estos valores se muestran mediante gráficas

### 3.1.2 Diseño

A continuación se van a describir las clases implementadas en Java y su utilidad:

- **Utils.java:** Es una clase pública que simplemente define variables estáticas que se usan a lo largo de todo el programa en otras clases. Por ejemplo, las rutas relativas de ficheros de entrada, ficheros de salida, la constante C, el porcentaje de error al crear el nuevo estado neuronal, el número de iteraciones en la parte de recuperación del patrón, etc.
- **ExcelHandler.java:** En esta clase es donde se lleva a cabo todo el preproceso previo a la creación de la red neuronal. El objetivo de esta clase es poder transformar el fichero excel a los valores requeridos (1, 0 y -1) en un array de dos dimensiones en el que la primera dimensión son los patrones y la segunda todas las leyes de cada uno de los países. Para ello, hay una primera función que escribe el resultado en un excel para una fácil comparación de los patrones y también escribe el resultado en un fichero que otra función utiliza para crear el array definitivo con los patrones y leyes ordenados.
- **RedNeuronal.java:** Es en esta clase donde está todo el funcionamiento de la red neuronal. En ella tiene lugar la fase de aprendizaje en la que se genera la matriz de pesos, así como la parte de recuperación en la que se genera el nuevo estado neuronal, la combinación lineal y se calculan los valores para los solapamientos de la red binaria y ternaria, que es el valor que se obtiene al final.

- **ChartPanelDemo.java:** Esta clase java dibuja las gráficas necesarias para la comparación en base a los valores del solapamiento obtenidos al ejecutar la red neuronal.
- **OpsMaths.java:** Aquí están todas las operaciones matemáticas que no tenía sentido incluir en RedNeuronal.java. Por ejemplo, multiplicación y suma de matrices, producto de vectores, función signo y función escalón para la función activación, comprobar si dos vectores son iguales.
- **Main.java:** Este es el programa principal desde donde se llaman a todas las funciones que hay en las distintas clases java, el resultado de la ejecución del main es la gráfica de los solapamientos.

## 4 Desarrollo

### 4.1 Preproceso

Este preproceso es la parte en la que se transforma el fichero excel (donde están todos los datos) en lo que van a ser los ficheros finales de entrada que sólo contienen los valores -1 y 1 para el caso de la red neuronal binaria y -1, 0 y 1 para la red ternaria.

Lo primero de todo es ver cómo se traducen los valores del excel a -1, 0 y 1.

Para el caso de la red ternaria es sencillo, si se encuentra con un 'Y' se asigna un 1, si se encuentra con un 'N' se asigna un -1 y para todo lo demás se asigna un 0. Hay dos excepciones en las cuales hay que especificar qué valor asignar y son "Economic group" y "Default Marital Property Regime". En la parte de [Anexos](#) se puede encontrar más información sobre qué significan y cómo se han tratado estos datos.

Para el caso de la red binaria se ha hecho un preproceso manual para aquellos valores que no coincidían con 'Y' y 'N'. Básicamente se han sustituido los valores no deseados por 'Y' y 'N' fijándose en qué ocurría ese mismo año para los países vecinos (con ayuda de Google Maps) del mismo continente. Se ha optado por hacer esta tarea de forma manual dado que los valores distintos de 'Y' y 'N' eran un porcentaje pequeño y la mayoría perteneciente a África y parte de Asia.

Además, se ha creado un fichero excel al final de este preproceso en el que los patrones están puestos en columnas (50 columnas para 50 patrones) para poder comparar los distintos patrones de una forma más sencilla porque el fichero de entrada para la red neuronal tiene los patrones por filas en vez de por columnas.

	Pais 1			Pais 2			Pais n		
	Ley 1	Ley 2	Ley n	Ley 1	Ley 2	Ley n	Ley 1	Ley 2	Ley n
Patron 1	1	1	-1	1	-1	1	1	-1	0
Patron 2	-1	0	1	1	-1	-1	-1	-1	1
Patron n	1	0	0	1	0	0	0	0	1

Tabla 4-1 Formato de entrada para la red neuronal

		Patron1	Patron2	Patron n
Pais 1	Ley1	1	-1	1
	Ley2	1	0	0
	Ley n	-1	1	0
Pais 2	Ley1	1	1	1
	Ley2	-1	-1	0
	Ley n	1	-1	0
Pais 3	Ley1	1	-1	0
	Ley2	-1	-1	0
	Ley n	0	1	1

**Tabla 4-2 Formato del fichero creado para la fácil comparación de patrones**

## 4.2 Fase de aprendizaje

En esta fase es donde se crea la matriz de pesos vista anteriormente en la parte del [Estado del Arte](#).

La matriz de pesos se va actualizando por cada nuevo patrón de entrada aprendido, por lo tanto hay que calcular la matriz de pesos por cada nuevo patrón y sumársela a la matriz de pesos del patrón anterior.

Además, al final de cada iteración se iguala la diagonal principal de la matriz a 0 debido a que en esta red todas las neuronas están conectadas entre sí menos consigo mismas.

Hay dos diferencias en la manera de calcular la matriz de pesos para la red bilineal y la bimodal. La primera es que para la bimodal las componentes de los patrones de entrada tienen que estar elevados al cuadrado, así que antes de calcular esta matriz de pesos se elevan al cuadrado en caso de calcular la matriz para la red bimodal. La segunda diferencia es que para la red bimodal, tal y como se ha visto en el Estado del Arte, primero se resta 'a' a cada una de las componentes antes de multiplicarse entre sí.

```

if (binariaOTernaria == Utils.Ternaria) {
    for (int i = 0; i < vectores.length; i++) {
        for (int j = 0; j < vectores[0].length; j++) {
            vectores[i][j] = (int) Math.pow(vectores[i][j], 2);
        }
    }
}

for (int i = 0; i < vectores.length; i++) {
    // Generar matriz pesos
    if(binariaOTernaria==Utils.Ternaria)
        matrizPesos = this.generarMatrizPesosTernaria(vectores[i], opsMaths, a[i]);
    else
        matrizPesos = this.generarMatrizPesos(vectores[i], opsMaths);

    matrizPesosResultados = opsMaths.sumaMatrices(matrizPesosResultados, matrizPesos);
    matrizPesosResultados = opsMaths.anulaDiagonalEnMatriz(matrizPesosResultados);
}

public double[][] generarMatrizPesos(int[] vector) {
    double[][] matrizResultado = new double[vector.length][vector.length];
    for (int i = 0; i < vector.length; i++)
        for (int j = 0; j < vector.length; j++)
            matrizResultado[i][j] = vector[i] * vector[j];
    return matrizResultado;
}

public double[][] generarMatrizPesosTernaria(int[] vector, double a) {
    double [][] matrizResultado = new double[vector.length][vector.length];
    for (int i = 0; i < vector.length; i++)
        for (int j = 0; j < vector.length; j++){
            double vectorI= (double) vector[i] - a;
            double vectorJ= (double) vector[j] - a;

            matrizResultado[i][j] = vectorI * vectorJ;
        }
    return matrizResultado;
}

```

Ilustración 4-1 Código Matriz pesos

## 4.3 Fase de recuperación

### 4.3.1 Cálculo nuevo estado neuronal

Lo primero que hay que hacer en la fase de recuperación es obtener un nuevo patrón (lo llamaremos estado neuronal) con el que se llegará a la conclusión de si la red lo ha recuperado o no, es decir, lo ha reconocido como uno de los patrones aprendidos.

Hay varias formas de calcular este nuevo estado neuronal, pero sólo se va a explicar la que se ha usado:

Se ha definido una variable estática con el porcentaje de ruido que se quiere usar en este nuevo estímulo inicial.

```
public static double porcentajeErrorIncluido= 0.1;
```

#### Ilustración 4-2 Código Porcentaje de error al crear el nuevo estado neuronal

Este porcentaje de error se multiplica por la longitud de los patrones de entrada y el resultado es el número de posiciones que hay que modificar. Así que se va a modificar este nuevo estado neuronal (cogiendo el último patrón aprendido como base) "numeroModificaciones" veces. La manera más fácil es modificar siempre las primeras posiciones.

```
public int[] generaNuevoEstadoNeuronal(int[] patronEntrada, int binariaOTernaria) {  
    int[] patronARrecuperar = patronEntrada;  
    int numeroModificaciones = (int) Math.round((patronEntrada.length * Utils.porcentajeErrorIncluido));  
  
    for (int i = 1; i <= numeroModificaciones; i++)  
        patronARrecuperar[i] = this.modificarPosicion(patronARrecuperar[i], binariaOTernaria);  
  
    return patronARrecuperar;  
}
```

#### Ilustración 4-3 Código creación del nuevo estado neuronal

Cuando la red es binaria la modificación de las posiciones es sencilla; si previamente había un 1 en esa posición ahora habrá un -1 y lo mismo al revés. Cuando la red es ternaria hay dos posibles opciones de modificación y la elegida será calculando un número aleatorio, es decir, si previamente había un 1, después va a haber un 0 ó un -1 de manera aleatoria.

### 4.3.2 Cálculo combinación lineal y función activación

Una vez calculado el nuevo estado neuronal se puede calcular la combinación lineal, que posteriormente se usará para calcular la función de activación.

Como se ha visto anteriormente, la combinación lineal no es más que la multiplicación del nuevo estado neuronal inicial por la matriz de pesos.

El código es igual para la red binaria y para la ternaria porque la matriz de Pesos ya ha sido previamente calculada con las componentes de los patrones elevados al cuadrado y restando la 'a' en el caso de la red ternaria.

```
public int[] multiplicaMatrizPorVector(int[][] matriz, int[] vector) {  
    int[] vectorResultado = new int[vector.length];  
    for (int i = 0; i < vector.length; i++)  
        for (int j = 0; j < matriz[i].length; j++)  
            vectorResultado[i] += vector[j] * matriz[i][j];  
    return vectorResultado;  
}
```

#### Ilustración 4-4 Código combinación lineal



Para calcular la función de activación diferenciamos entre la función escalón (red bimodal), la función signo (red binaria) y la función combinada (red ternaria), que como ya se ha visto anteriormente en la Ecuación 2-6, es la multiplicación de ambas.

```

public int[] calculaVectorActivacion(int[] vector,
| int binariaOTernaria) {
    //bicuadratica
    if (binariaOTernaria == Utils.Bicuadratica){
        for (int i = 0; i < vector.length; i++){
            vector[i] = this.funcionEscalon(vector[i]);
        }
        return vector;
    }
    //Ternaria
    else if(binariaOTernaria == Utils.Ternaria){
        for (int i = 0; i < vector.length; i++){
            vector[i] = this.funcionCombinada(vector[i]);
        }
        return vector;
    }
    //binaria
    for (int i = 0; i < vector.length; i++){
        vector[i] = this.funcionSigno(vector[i]);
    }
    return vector;
}

public int funcionCombinada (double numero){
    int h=0;
    h=this.funcionModulo(numero);
    return ((this.funcionSigno(numero))
        * (this.funcionEscalon(numero+h)));
}

public int funcionSigno(double numero) {
    if (numero < 0)
        return -1;
    return 1;
}

public int funcionEscalon(double numero) {
    if (numero <= 0)
        return 0;
    return 1;
}

public int funcionModulo (double numero){
    if (numero<0)
        numero=numero*-1;
    return numero;
}

```

**Ilustración 4-5 Código función activación y funciones signo, escalón, módulo y combinada.**

Sabiendo cómo se calcula la función de activación sólo falta describir cómo se llega al punto de estabilidad descrito en el Estado del Arte. Para llegar a ese punto de estabilidad es necesario calcular la combinación lineal para el instante  $t$  y ver si coincide con la del instante  $t+1$ , en ese caso el algoritmo para. De no ser así se repiten estos pasos con tiempos posteriores hasta llegar el punto de estabilidad o alcanzar el máximo número de iteraciones.

```

vectorCombinacionLineal = opsMaths.multiplicaMatrizPorVector(matrizPesos, vectorARecuperar);
vectorH = opsMaths.calculaVectorActivacion(vectorCombinacionLineal, binariaOTernaria);

for (int i = 0; i < Utils.numeroIteraciones; i++) {
    vectorCombinacionLinealAux = opsMaths.multiplicaMatrizPorVector(matrizPesos, vectorH);
    vectorHAux = opsMaths.calculaVectorActivacion(vectorCombinacionLinealAux, binariaOTernaria);

    if (opsMaths.compruebaVectoresIguales(vectorHAux, vectorH) == true)
        break;

    vectorH = vectorHAux;
}

public boolean compruebaVectoresIguales(int[] vector1, int[] vector2) {
    for (int i = 0; i < vector1.length; i++)
        if (vector1[i] != vector2[i])
            return false;
    return true;
}

```

**Ilustración 4-6 Código alcanzar punto de estabilidad**

## 4.4 Estudio del éxito de la recuperación

Para saber cómo de exitosa es la recuperación de estos patrones hay que calcular los valores de los solapamientos descritos en el [Estado del Arte](#).

La manera de obtener 'm' para la red binaria y 'n' para la ternaria (las fórmulas son iguales) es sumando el acumulado de la multiplicación de los valores de entrada de las neuronas por el valor del vector de activación. A ese resultado se le divide la multiplicación del número de neuronas multiplicado por 'a'.

```
public double calculaSolapamiento(double a, double[] vectorH, int[] vectorXi, double a1) {  
    double solapamiento = 0.0;  
    double aPorN = 0.0;  
  
    for (int i = 0; i < vectorXi.length; i++)  
        solapamiento += vectorXi[i] * vectorH[i];  
  
    aPorN = vectorXi.length * a1;  
    solapamiento = solapamiento / aPorN;  
  
    return solapamiento;  
}
```

Ilustración 4-7 Código solapamiento 'm' y 'n'

La única diferencia es que para la red binaria el valor de 'a' va a ser el 100% y para la red ternaria hay que calcularlo sumando todos los valores del patrón (esto funciona porque estos valores se han elevado previamente al cuadrado, así que sólo va a haber 1s y 0s) y dividiéndolo por el número total de neuronas.

```
public double[] calculaA(int binariaOTernaria, int[][] vectores) {  
    double resultadoA = 0;  
    double[] a = new double[vectores.length];  
  
    if (binariaOTernaria == Utils.Binaria) {  
        for (int i = 0; i < vectores.length; i++)  
            a[i] = 1.0;  
        return a;  
    }  
  
    for (int i = 0; i < vectores.length; i++) {  
        resultadoA = 0.0;  
        for (int j = 0; j < vectores[0].length; j++)  
            resultadoA += vectores[i][j];  
        a[i] = resultadoA / vectores[0].length;  
    }  
    return a;  
}
```

Ilustración 4-8 Código cálculo de 'a'

La última variable que se necesita para calcular el valor del solapamiento ternario es 'q' que se calcula como el sumatorio de la salida de la función activación dividido entre el número total de neuronas.

```
public double calculaQ(int binariaOTernaria, double[] vectorFuncionActivacion) {  
    double suma = 0.0;  
    for (int i = 0; i < vectorFuncionActivacion.length; i++)  
        suma += vectorFuncionActivacion[i];  
  
    return suma / vectorFuncionActivacion.length;  
}
```

**Ilustración 4-9 Código cálculo de 'q'**

Una vez ya se tienen todos los valores anteriores, ya se puede calcular el solapamiento ternario 'l'.

```
public double calculaQ(int binariaOTernaria, double[] vectorFuncionActivacion) {  
    double suma = 0.0;  
    for (int i = 0; i < vectorFuncionActivacion.length; i++)  
        suma += vectorFuncionActivacion[i];  
  
    return suma / vectorFuncionActivacion.length;  
}
```

**Ilustración 4-10 Código cálculo de 'l'**

## 4.5 Elaboración de gráficas para la comparación

Una vez ejecutada por completo la red neuronal, se van a obtener como resultado dos arrays (red bilineal y bimodal) con valores que oscilan entre [-1 y 1] con los que se puede hacer la combinación para obtener la red ternaria.

Son estos dos arrays los que sirven de entrada a la clase que dibuja las gráficas creando un JPanel que es rellenado con el set de datos recibido.

```
public ChartPanelDemo(double[] estudioRecuperacionBilineal, double[] estudioRecuperacionBimodal) {
    super("Estudio Sistema neuronal Ternario");

    JPanel chartPanel = createChartPanel(estudioRecuperacionBilineal, estudioRecuperacionBimodal);
    add(chartPanel, BorderLayout.CENTER);
    setSize(640, 480);
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    setLocationRelativeTo(null);
}

private JPanel createChartPanel(double[] estudioRecuperacionBilineal, double[] estudioRecuperacionBimodal) {

    String chartTitle = "Estudio Sistema neuronal Ternario";
    String xAxisLabel = "Número patrones";
    String yAxisLabel = "Solapamiento";

    XYDataset dataset = createDataset(estudioRecuperacionBilineal, estudioRecuperacionBimodal);
    JFreeChart chart = ChartFactory.createXYLineChart(chartTitle, xAxisLabel, yAxisLabel, dataset);
    XYPlot plot = chart.getXYPlot();

    XYLineAndShapeRenderer renderer = new XYLineAndShapeRenderer();
    plot.setRenderer(renderer);

    return new ChartPanel(chart);
}

private XYDataset createDataset(double[] estudioRecuperacionBilineal, double[] estudioRecuperacionBimodal) {
    XYSeriesCollection dataset = new XYSeriesCollection();
    XYSeries series1 = new XYSeries("Solapamiento red bilineal \"m\"");
    XYSeries series2 = new XYSeries("Solapamiento red bimodal \"l\"");

    for (int i=0; i<estudioRecuperacionBilineal.length; i++)
        series1.add(i, estudioRecuperacionBilineal[i]);
    dataset.addSeries(series1);
    for (int i=0; i<estudioRecuperacionBimodal.length; i++)
        series2.add(i, estudioRecuperacionBimodal[i]);
    dataset.addSeries(series2);

    return dataset;
}
```

Ilustración 4-11 Código generación gráficas

# 5 Integración, pruebas y resultados

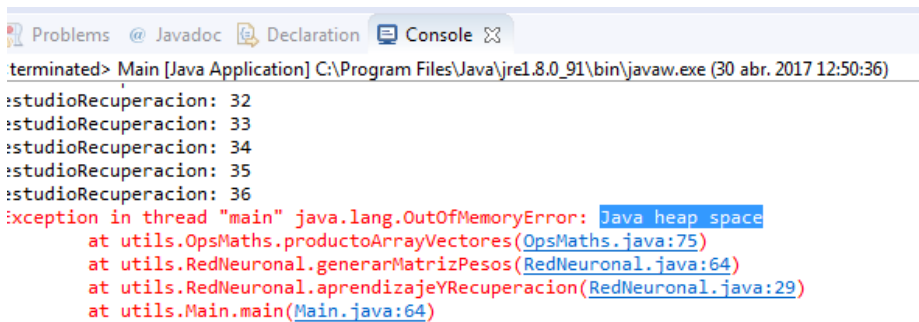
---

## 5.1 Introducción

En esta sección se habla de los distintos resultados que se han obtenido.

Los datos usados para realizar estas pruebas han sido un total de 50 patrones con 1800 atributos cada patrón (18 leyes por cada uno de los 100 países) para las pruebas con patrones reales y 200 patrones con 1800 atributos cada patrón para las pruebas con patrones aleatorios.

Por supuesto las pruebas unitarias realizadas para la comprobación del correcto funcionamiento de los algoritmos han sido realizadas con unos datos de entrada mucho menores (3-4 patrones con 4-5 atributos cada patrón), puesto que la correcta comprobación de los algoritmos de suma, resta, etc. de matrices se hace inmanejable. No obstante, es necesario realizar pruebas con los datos que vamos a usar en el futuro, pues al tratar con arrays de dos y tres dimensiones y 1800 atributos en cada dimensión es posible que en algún momento veamos que tenemos que esperar mucho (falta de rendimiento) o que incluso las operaciones no se puedan realizar (falta de memoria en Eclipse) como se puede ver en la siguiente foto.

The image shows a screenshot of the Eclipse IDE's console window. The title bar at the top indicates the active tab is 'Console'. The console output shows the execution of a Java application. It starts with a 'terminated>' message, followed by several lines of output: 'Main [Java Application] C:\Program Files\Java\jre1.8.0\_91\bin\javaw.exe (30 abr. 2017 12:50:36)', ':studioRecuperacion: 32', ':studioRecuperacion: 33', ':studioRecuperacion: 34', ':studioRecuperacion: 35', and ':studioRecuperacion: 36'. Finally, an exception is thrown: 'Exception in thread "main" java.lang.OutOfMemoryError: Java heap space'. The stack trace follows, showing the error occurred in 'utils.OpsMaths.productoArrayVectores' at line 75, which was called by 'utils.RedNeuronal.generarMatrizPesos' at line 64, which was called by 'utils.RedNeuronal.aprendizajeYRecuperacion' at line 29, which was called by 'utils.Main.main' at line 64. The file names and line numbers are highlighted in blue in the original image.

```
terminated> Main [Java Application] C:\Program Files\Java\jre1.8.0_91\bin\javaw.exe (30 abr. 2017 12:50:36)
:studioRecuperacion: 32
:studioRecuperacion: 33
:studioRecuperacion: 34
:studioRecuperacion: 35
:studioRecuperacion: 36
Exception in thread "main" java.lang.OutOfMemoryError: Java heap space
    at utils.OpsMaths.productoArrayVectores(OpsMaths.java:75)
    at utils.RedNeuronal.generarMatrizPesos(RedNeuronal.java:64)
    at utils.RedNeuronal.aprendizajeYRecuperacion(RedNeuronal.java:29)
    at utils.Main.main(Main.java:64)
```

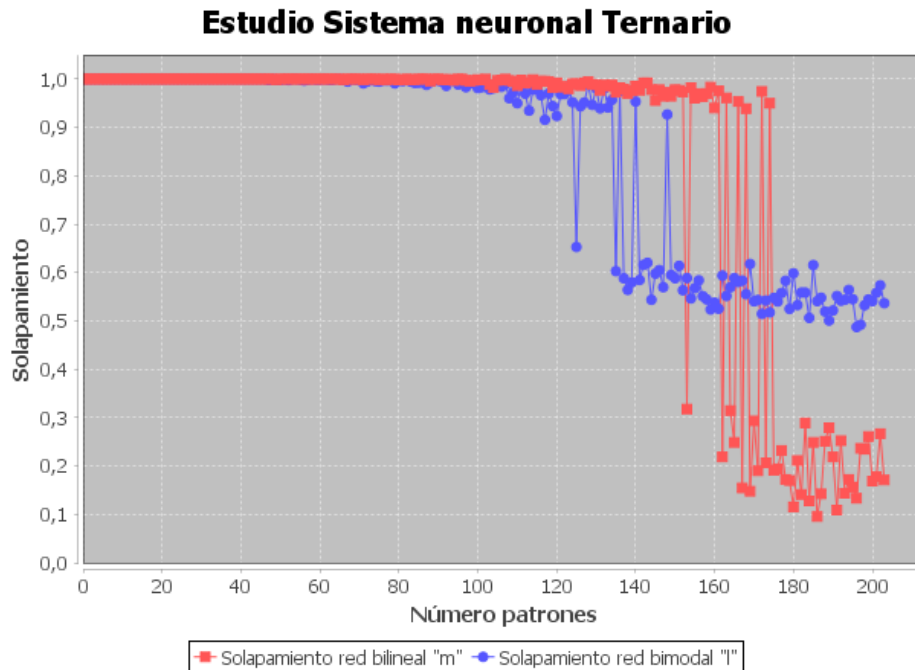
**Ilustración 5-1 Falta de memoria en Eclipse**

Al recibir el error anterior se modificaron los algoritmos, y el resultado final es que la memoria consumida por Eclipse en las ejecuciones no llega a 1 GB. No obstante, en cada ejecución se genera esa gráfica, y cada una de esas gráficas ocupa una media de 1.2 GB.

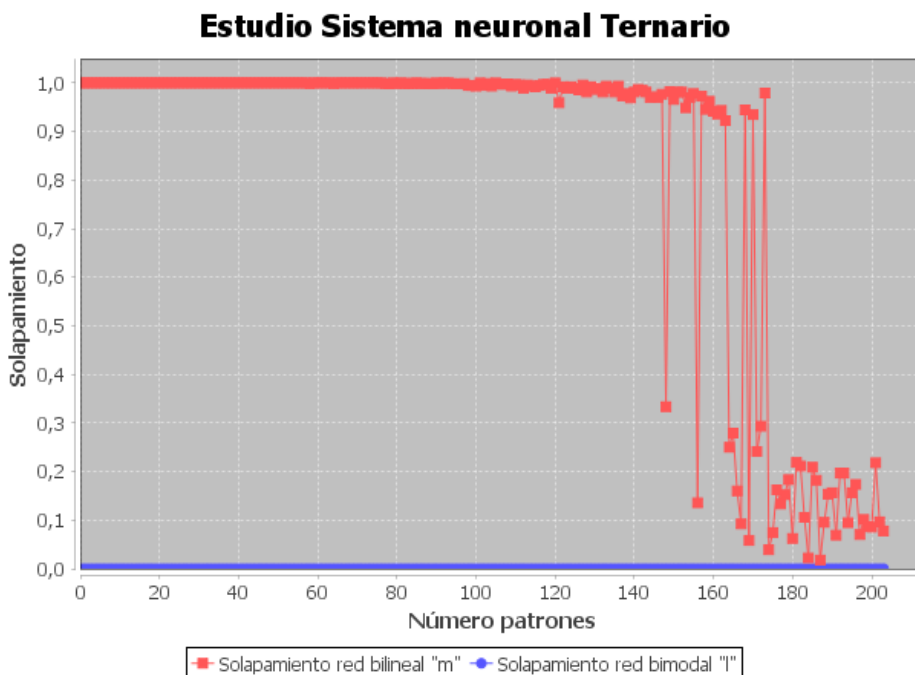
Nombre de imagen	Nombre ...	CPU	Memoria (espacio de traba...	Descripción
Discord.exe *32	HugGarcia	00	19.760 KB	Discord
dwm.exe	HugGarcia	01	15.036 KB	Administr...
eclipse.exe	HugGarcia	00	913.056 KB	eclipse.exe
ETDCtrl.exe	HugGarcia	00	420 KB	ETD Cont...
ETDCtrlHelper.exe	HugGarcia	00	256 KB	ETD Cont...
EvernoteClipper.exe *32	HugGarcia	00	292 KB	Evernote ...
explorer.exe	HugGarcia	00	22.576 KB	Explorado...
firefox.exe	HugGarcia	00	7.916 KB	Firefox
firefox.exe	HugGarcia	00	162.928 KB	Firefox
firefox.exe	HugGarcia	00	564.936 KB	Firefox
IASStorIcon.exe *32	HugGarcia	00	1.956 KB	IASStorIcon
igfxhik.exe	HugGarcia	00	276 KB	igfxhik Mo...
javaw.exe	HugGarcia	00	1.188.996 KB	Java(TM) ...
javaw.exe	HugGarcia	00	1.232.164 KB	Java(TM) ...
javaw.exe	HugGarcia	00	1.238.340 KB	Java(TM) ...
javaw.exe	HugGarcia	00	1.277.460 KB	Java(TM) ...
javaw.exe	HugGarcia	00	1.234.488 KB	Java(TM) ...
javaw.exe	HugGarcia	00	1.237.364 KB	Java(TM) ...
jucheck.exe *32	HugGarcia	00	1.140 KB	Java Upd...
jusched.exe *32	HugGarcia	00	1.116 KB	Java Upd...
NetworkManager.exe *32	HugGarcia	00	4.268 KB	NetworkM...
NvBackend.exe *32	HugGarcia	00	2.516 KB	NVIDIA G...
nvtray.exe	HugGarcia	00	348 KB	NVIDIA S...
nvsvsc.exe		00	380 KB	
nvxdsync.exe		00	428 KB	
RdrCEF.exe *32	HugGarcia	00	1.132 KB	Adobe Rd...
RdrCEF.exe *32	HugGarcia	00	3.484 KB	Adobe Rd...

**Ilustración 5-2 Memoria usada por eclipse en las ejecuciones y por las gráficas generadas**

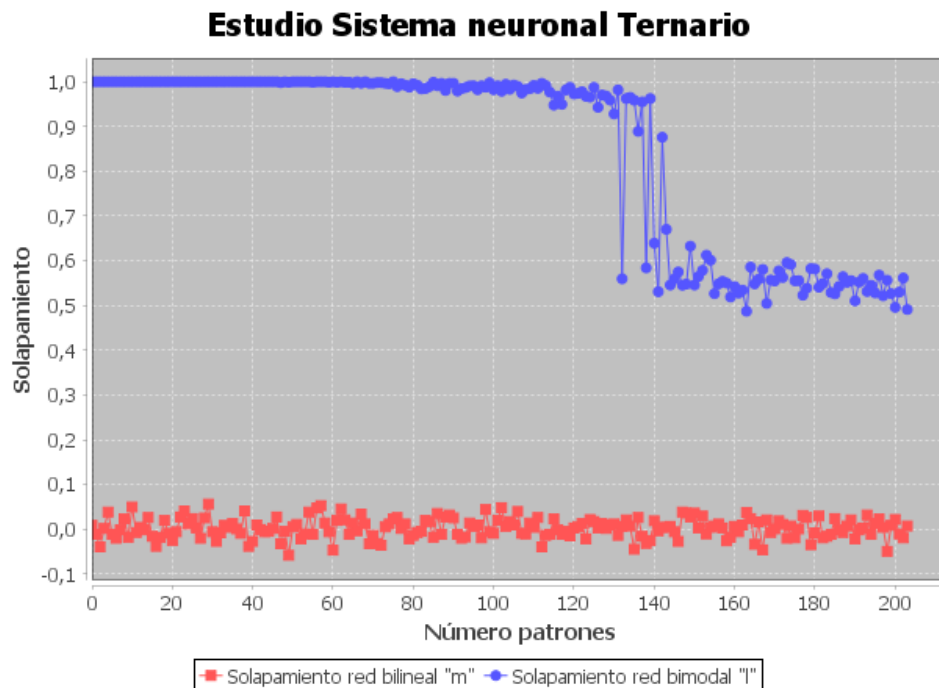
## 5.2 Resultados



**Ilustración 5-3 Solapamiento para patrones aleatorios con  $C=0.5$ , número de iteraciones=500, 25% de ruido, tiempo de ejecución total 5 minutos y  $D=1$**



**Ilustración 5-4 Solapamiento para patrones aleatorios con  $C=1$ , número de iteraciones=500, 25% de ruido, tiempo de ejecución total 2 minutos y 20 segundos y  $D=1$**



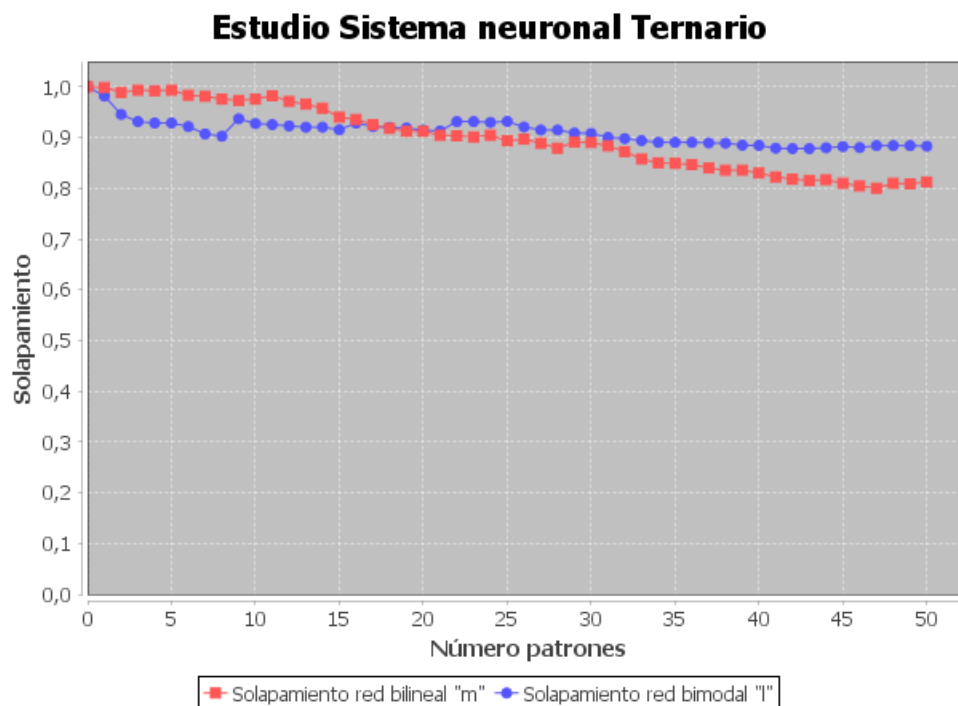
**Ilustración 5-5 Solapamiento para patrones aleatorios con  $C=0$ , número de iteraciones=500, 25% de ruido, tiempo de ejecución total 3 minutos y  $D=1$**

Las tres gráficas anteriores muestran el resultado de aplicar el algoritmo descrito a lo largo del documento a 200 patrones aleatorios con distintos valores de  $C$ . Estas pruebas han sido las primeras que se han hecho puesto que por experiencias anteriores (gracias a mi tutor, David) se sabe que dependiendo de qué patrones de entrada se usen puede haber distintos comportamientos. Por lo tanto, estos resultados son simples resultados de test y validación del algoritmo.

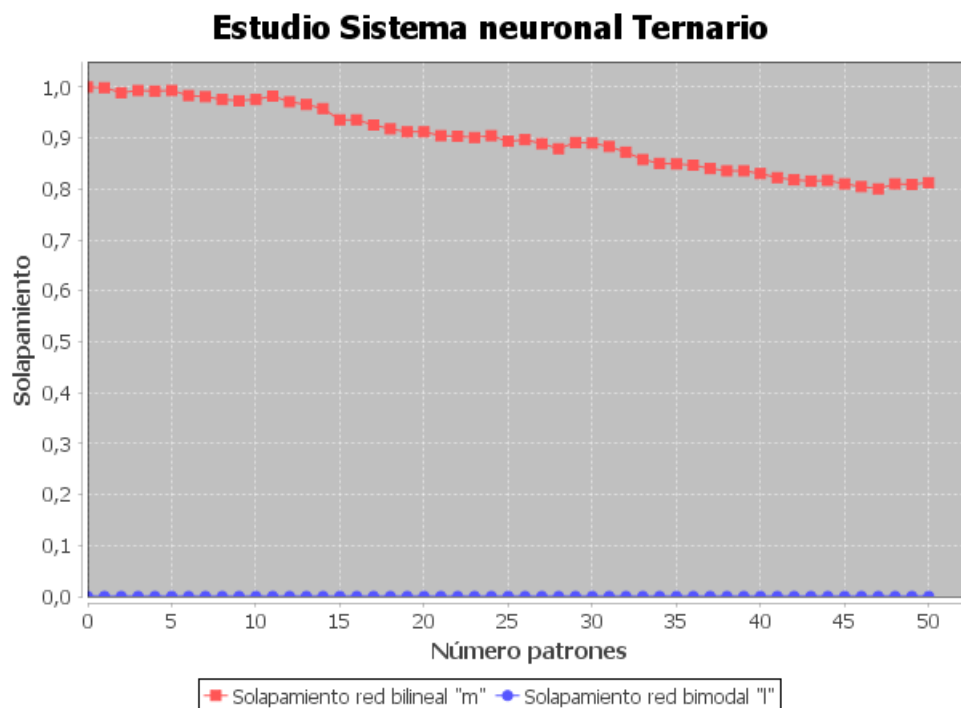
Las pruebas resultantes con los patrones aleatorios confirman que el algoritmo está bien desarrollado puesto que se ve que, en un momento determinado, la gráfica (el solapamiento) cae. Esta caída ocurre de una manera un poco extraña pero fácilmente demostrable al estar usando patrones aleatorios en los que los valores generados difieren mucho unos de otros. Esto también explica por qué el número máximo de iteraciones (500 en este caso) es alcanzado a partir del patrón número 100.



Después de comprobar que el algoritmo funciona para los patrones aleatorios es cuando se pueden extraer las pruebas con los patrones reales de género.

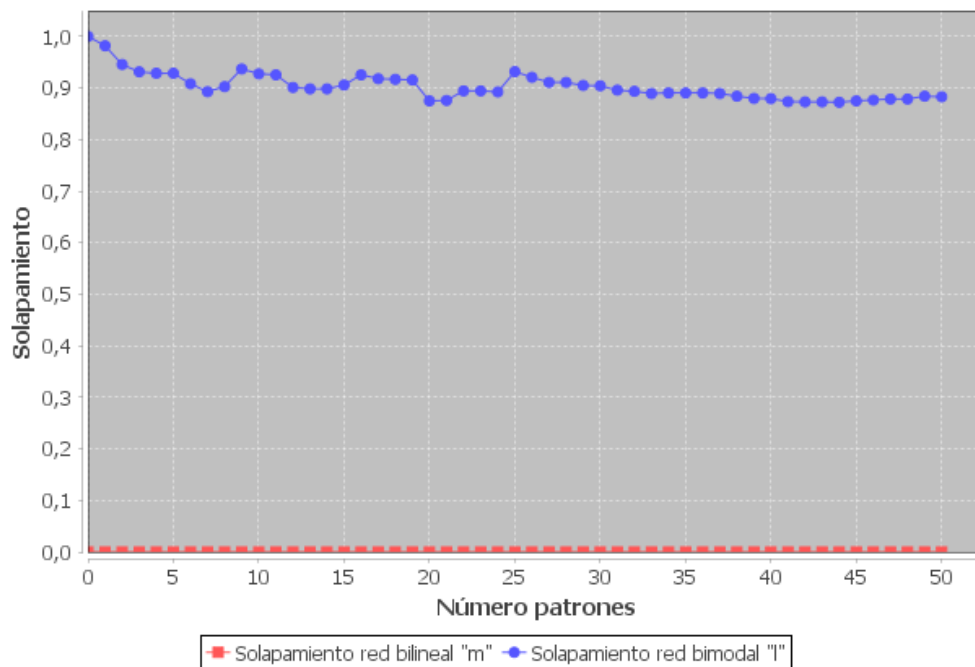


**Ilustración 5-6 Solapamientos para patrones reales con  $C=0.5$ , número de iteraciones =1000, 25% de ruido, tiempo ejecución total 5 segundos y  $D=1$**



**Ilustración 5-7 Solapamiento para patrones reales con  $C= 1$ , número de iteraciones = 1000, 25% de ruido, tiempo de ejecución total 5 segundos y  $D=1$**

### Estudio Sistema neuronal Ternario



**Ilustración 5-8 Solapamiento para patrones reales con  $C=0$ , número de iteraciones= 1000, 25% de ruido, tiempo de ejecución total 5 segundos y  $D=1$**

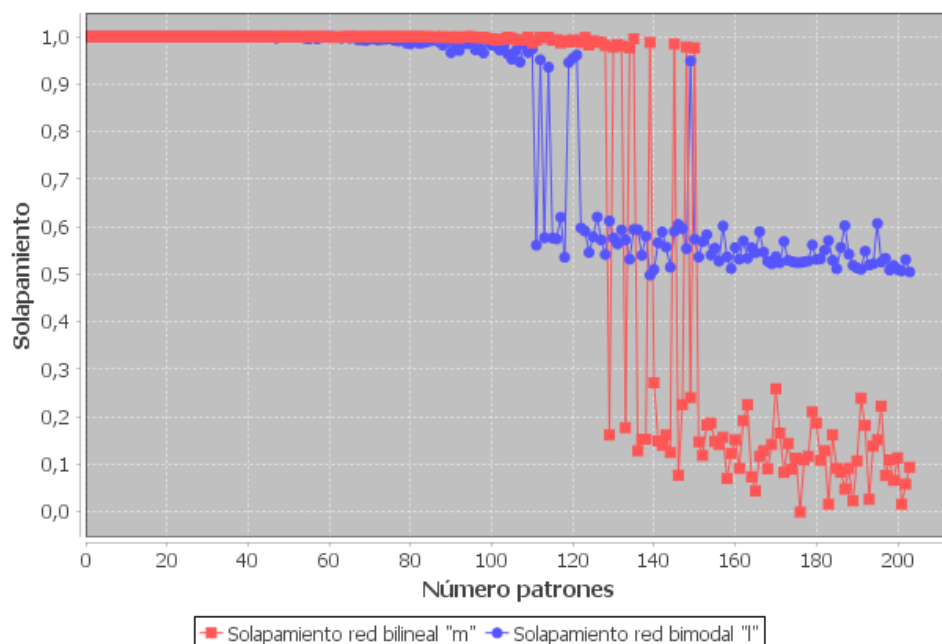
Aquí, como se había previsto anteriormente, el comportamiento es distinto.

Por una parte los tiempos de ejecución son muy pequeños en comparación con los patrones aleatorios, esto es porque apenas hay iteraciones en la parte de la recuperación. Además, en estas gráficas no se observa la caída de la que se había hablado y observado antes. Esto es debido a que esa caída ocurría en el patrón 120 más o menos y aquí sólo se cuentan con 50. Hay una manera para "emular" tener más patrones, y es reduciendo el número de conexiones entre neuronas, esto se consigue incrementando 'D'. Aún así, las gráficas se mantienen iguales.

Todas estas peculiaridades son debido a que los patrones están muy correlacionados entre sí, es decir, los valores de los patrones tienen un porcentaje de variación muy pequeño entre ellos.

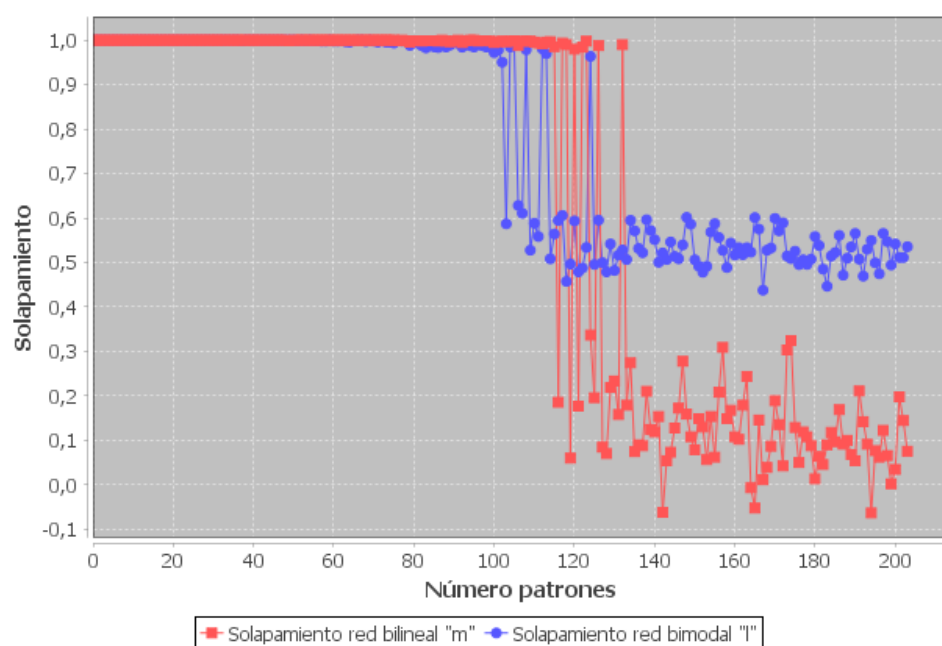
Las siguientes gráficas contienen las pruebas de haber ejecutado el algoritmo con los patrones aleatorios incrementando 'D' para que quede demostrado que se puede "emular" tener más patrones para que la gráfica caiga antes.

### Estudio Sistema neuronal Ternario



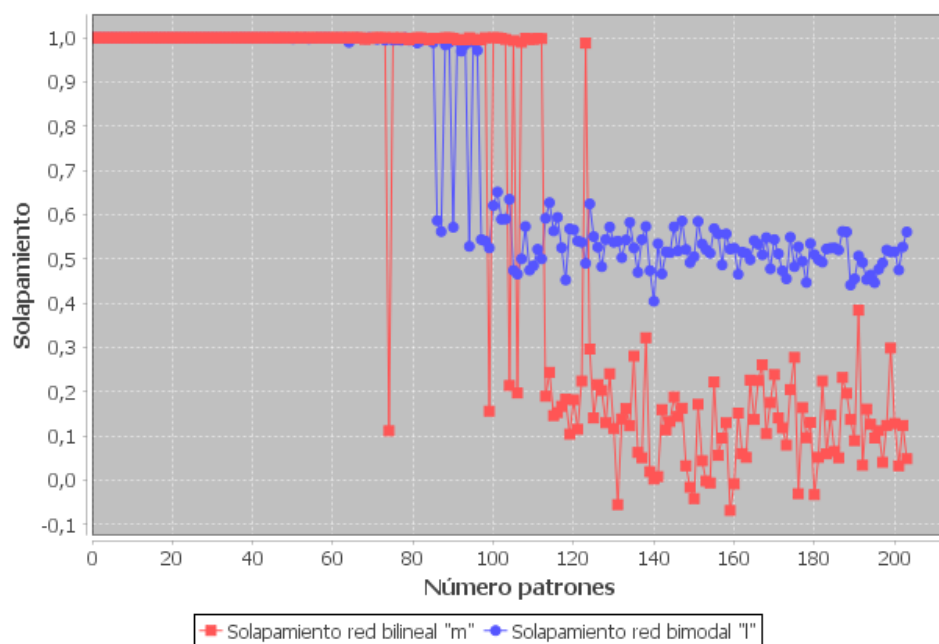
**Ilustración 5-9 Solapamiento para patrones aleatorios con  $C=0,5$ , número de iteraciones=500 y 25% de ruido. Tiempo de ejecución total 5 minutos y  $D=10$**

### Estudio Sistema neuronal Ternario



**Ilustración 5-10 Solapamiento para patrones aleatorios con  $C=0,5$ , número de iteraciones=500 y 25% de ruido. Tiempo de ejecución total 5 minutos y  $D=20$**

### Estudio Sistema neuronal Ternario



**Ilustración 5-11 Solapamiento para patrones aleatorios con  $C=0,5$ , número de iteraciones=500 y 25% de ruido. Tiempo de ejecución total 5 minutos y  $D=40$**

## 6 Conclusiones y trabajo futuro

---

### 6.1 Conclusiones

De las fórmulas y datos anteriormente descritos se puede observar que el solapamiento 'm' no cuenta con 0s y que sólo distingue entre signos.

Por otra parte, el solapamiento 'l' sólo distingue entre si hay dato (1) o no (0).

Es por ello que se llega a la conclusión de que hay que juntar ambos solapamientos para obtener información más robusta, y, basándose en los resultados obtenidos, la conclusión es que, para este específico caso de igualdad de género, la red ternaria aporta esa información más robusta sin perder capacidad de recuperación.

Se ha visto incluso que, para patrones reales, el solapamiento bimodal 'l' incluso sube por encima del solapamiento binario 'm' a partir de un cierto número de patrones, por lo que la combinación de ambas ofrece robustez.

Esto confirma la pregunta que se lanzaba en la parte de Introducción donde se dudaba si para este caso específico también la red ternaria funcionaba mejor que la binaria.

### 6.2 Trabajo futuro

Cualquier gran proyecto (como este trabajo de fin de grado) es casi siempre mejorable o ampliable, y en el descrito a lo largo de este documento hay algunos aspectos a tener en cuenta para posibles implementaciones futuras o simplemente siguientes pasos que se quieran seguir:

- En el apartado de [Resultados](#) se ha visto que los patrones de género poseen mucha correlación entre sí. Una de las posibles líneas de trabajo futuras puede ser evitar esta correlación haciendo uso de la regla pseudoinversa en la que la matriz de pesos es calculada de otra forma distinta. [17]
- En esta implementación se ha elegido usar los valores de 1 y -1 en el caso de que exista valor en el fichero excel y un 0 en el caso de que no exista valor y/o no aplique. Esta elección está basada en que el número 0 es el intervalo medio entre los otros dos valores, y por lo tanto podría ser tanto 1 como -1. Una posible implementación alternativa sería cambiar esta elección y hacer que el número "nulo" sea el -1, de esta manera la red contaría con mayor cantidad de 0s. A priori, no habría ninguna ventaja en esta implementación, puesto que, en la presuposición que se hace, la distancia entre los posibles valores y el valor nulo puede ser 1 ó 2, mientras que con la elección del valor nulo 0 la distancia es siempre 1. No obstante, puede ser que para este determinado caso en la que los patrones están muy correlacionados se puedan observar algunos resultados interesantes.
- Otro próximo paso será mantener el contacto con la facultad de económicas para ver si para ellos es interesante usar este trabajo de fin de grado y el programa generado como un proyecto docente para ellos.

- Los patrones utilizados en esta implementación han sido los años en los que los valores de las neuronas eran las 18 leyes de cada país, es decir, 51 patrones con 1800 atributos. Queda abierta una nueva implementación en la que se usarían los 100 países como patrones y como los valores de las neuronas las leyes de cada año, es decir, 100 patrones con 918 atributos.
- Se ha hecho la implementación de una red neuronal ternaria en la que se cuenta con tres estados en la que se le da más exactitud al añadir otro estado para los valores que se desconozcan. Una de las posibles implementaciones (mucho más difícil y compleja que las descritas anteriormente) sería añadir otro estado más, es decir, una red cuaternaria. Desde el punto de vista teórico habría que añadir potencias a las ecuaciones ya existentes.
- Publicación en revista científica (Quality and Quantity, por ejemplo) del trabajo realizado atendiendo a posibles pequeñas mejoras o matices como los descritos anteriormente.

## Referencias

---

- [1] <http://puntogenero.inmujeres.gob.mx/madig/igualdad/index.html>
- [2] [https://es.wikipedia.org/wiki/Igualdad\\_de\\_g%C3%A9nero](https://es.wikipedia.org/wiki/Igualdad_de_g%C3%A9nero)
- [3] <http://eige.europa.eu/sites/default/files/documents/mh0215616enn.pdf>
- [4] <http://www.telesurtv.net/telesuragenda/Por-que-es-importante-la-igualdad-de-genero-20170227-0039.html>
- [5] <http://www.undp.org/content/undp/es/home/sustainable-development-goals/goal-5-gender-equality.html>
- [6] [http://mueveteporlaigualdad.org/docs/educacion\\_en\\_igualdad\\_castellano.pdf](http://mueveteporlaigualdad.org/docs/educacion_en_igualdad_castellano.pdf)
- [7] [http://www.eldiario.es/piedrasdepapel/Genero-domestico-Tiende-Espana-igualdad\\_6\\_345125504.html](http://www.eldiario.es/piedrasdepapel/Genero-domestico-Tiende-Espana-igualdad_6_345125504.html)
- [8] <http://www.elfinanciero.com.mx/economia/igualdad-de-genero-economica-y-laboral-una-realidad-en-170-anos-wef.html>
- [9] <https://www.weforum.org/es/agenda/2015/11/informe-global-de-la-brecha-de-genero-2015/>
- [10] <http://www.datosmacro.com/demografia/indice-brecha-genero-global>
- [11] [https://www.fro.utn.edu.ar/repositorio/catedras/quimica/5\\_anio/orientadora1/monografias/matich-redesneuronales.pdf](https://www.fro.utn.edu.ar/repositorio/catedras/quimica/5_anio/orientadora1/monografias/matich-redesneuronales.pdf)
- [12] [https://es.wikipedia.org/wiki/Memoria\\_asociativa\\_\(RNA\)](https://es.wikipedia.org/wiki/Memoria_asociativa_(RNA))
- [13] <https://es.wikipedia.org/wiki/BRICS>
- [14] <https://en.wikipedia.org/wiki/Triadization>
- [15] <https://www.ch.ch/en/matrimonial-regime/>
- [16] <https://www.bbva.com/es/poder-adquisitivo-todos/>
- [17] <http://bibliotecadigital.univalle.edu.co/bitstream/10893/8865/1/CB-0516256-LA%20PSEUDOINVERSA%20EN%20EL%20PROCESO%20DE%20APRENDIZAJE%20DEL%20ASOCIADOR%20LINEAL.pdf>
- [18] <http://www.telegraph.co.uk/travel/maps-and-graphics/mapped-the-best-and-worst-countries-for-gender-equality/>
- [19] [http://4.bp.blogspot.com/-\\_5Od0TTedgU/VcPw3Piw3Pi3I/AAAAAAAAAfo/vSCGeIyvs5c/s1600/Neurona%2BBiologica.png](http://4.bp.blogspot.com/-_5Od0TTedgU/VcPw3Piw3Pi3I/AAAAAAAAAfo/vSCGeIyvs5c/s1600/Neurona%2BBiologica.png)
- [20] [https://c.mql5.com/2/12/Fig5\\_Hopfield\\_neural\\_net.png](https://c.mql5.com/2/12/Fig5_Hopfield_neural_net.png)

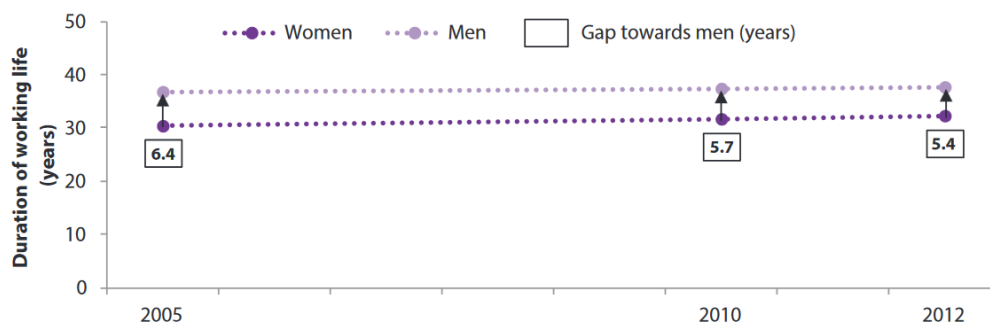
## Glosario

---

<b>UE</b>	Unión Europea
<b>ISCED</b>	Clasificación Internacional Normalizada de la Educación (español)
<b>PNUD</b>	Programa de las Naciones Unidas para el Desarrollo
<b>PPS</b>	Purchasing Power Standard (Estándar de Poder Adquisitivo). En la parte de Anexos se puede encontrar una definición con más detalle.

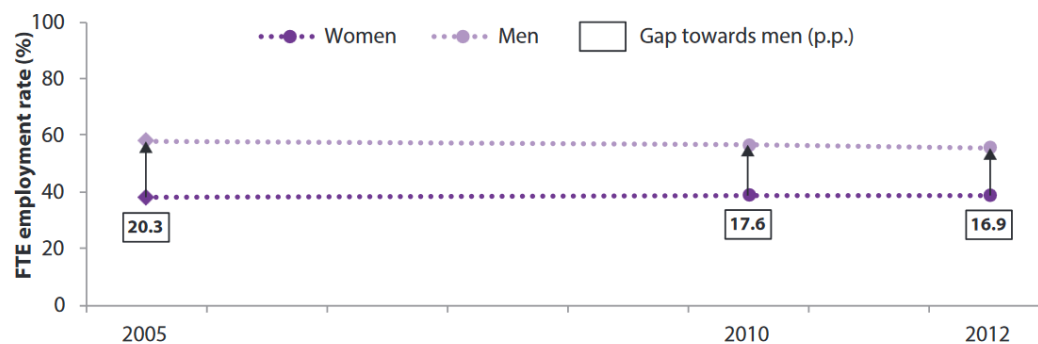
## Anexos

### A Gráficas de la desigualdad de género



Source: Eurostat, LSF (lfsi\_dwl\_a).

**Ilustración A-0-1 Duración de la vida laboral por sexo en la UE [3]**

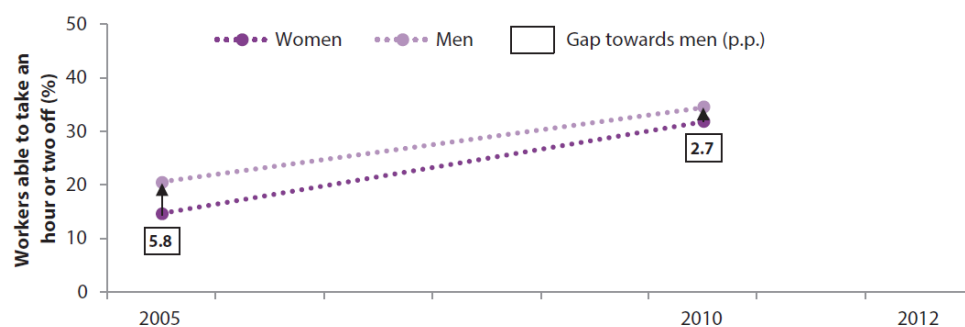


Source: Eurostat, LSF (data calculated by Eurostat at EIGE's request).

Note: Due to lack of data availability for the EU-28 in 2005, the EU-27 average was used.

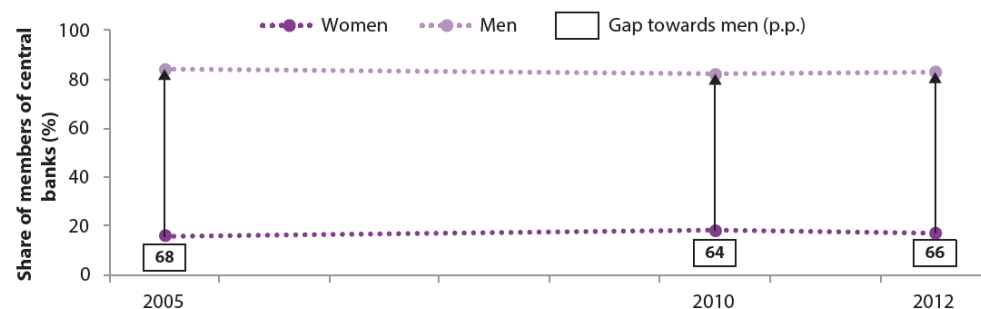
**Ilustración A-0-2 Porcentaje de empleo a jornada completa por sexo en la UE [3]**





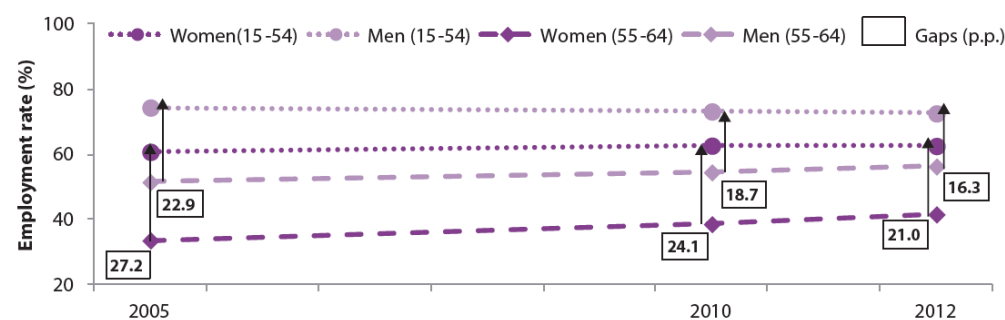
Source: Eurofound, European Working Conditions Survey (Q43).

**Ilustración A-0-3 Trabajadores que son capaces de cogerse una o dos horas durante el horario de trabajo para atender asuntos personales o familiares por sexo en la UE [3]**



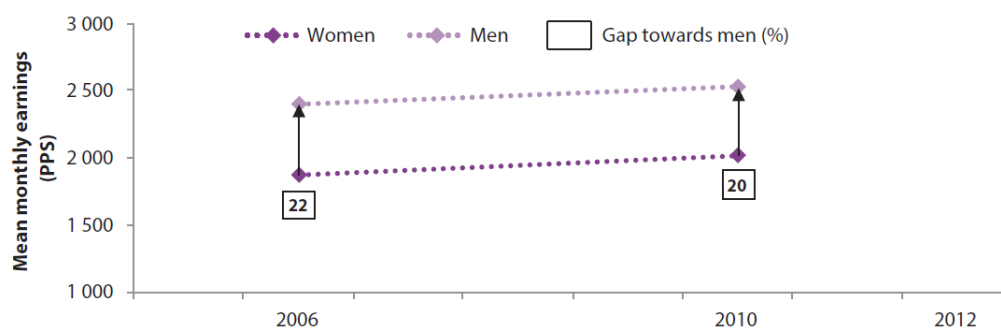
Source: EC-DG Justice, WMID.

**Ilustración A-0-4 Miembros del banco central por sexo en la UE [3]**



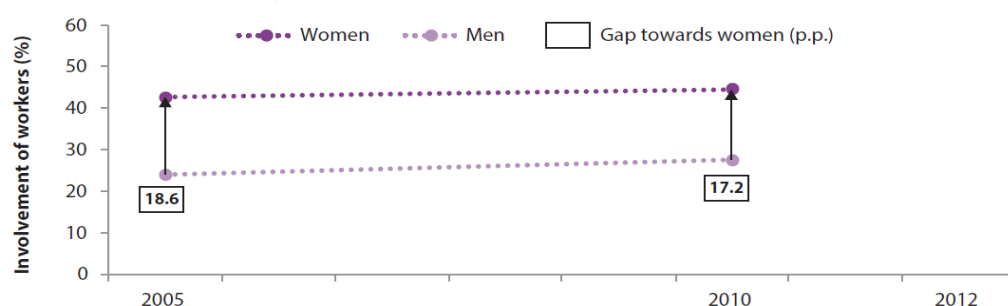
Source: Eurostat, LFS (Ifsa\_ergacob).

**Ilustración A-0-5 Tasas de empleo de los trabajadores entre 15-54 y de los trabajadores entre 55-64 por sexo en la UE [3]**



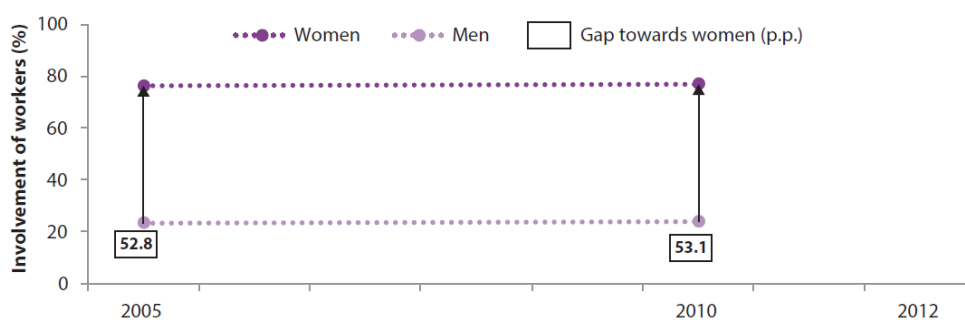
Source: Eurostat, SES (earn\_ses10\_20).  
Note: EU-28 average was not available for 2006 and 2010; the EU-27 average was used.

### Ilustración A-0-6 Ganancia media mensual (PPS) por sexo en la UE [3]



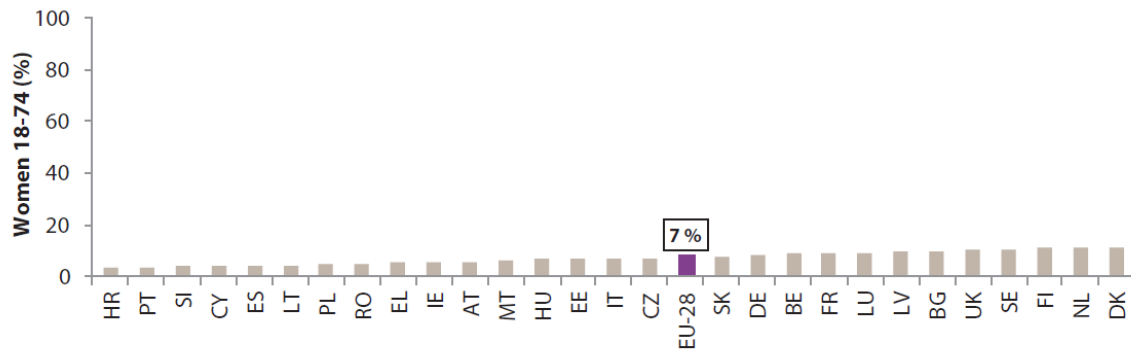
Source: Eurostat, EWCS (ef2c for 2010, ef4c for 2005).

### Ilustración A-0-7 Participación de los trabajadores en el cuidado y la educación de los niños y/o nietos, todos los días durante una hora o más, por sexo en la UE [3]



Source: Eurostat, EWCS (ef2d for 2010, ef4d for 2005).

### Ilustración A-0-8 Participación de los trabajadores en la cocina y las tareas domésticas, todos los días durante una hora o más, por sexo en la UE [3]



Source: FRA (2014a), EU-wide Survey on Violence against Women database (DVS\_A09).

**Ilustración A-0-9 Porcentaje de mujeres que han sufrido violencia sexual por su pareja en 2012 [3]**

## B Excepciones en el preproceso de los datos desde el fichero excel

A continuación se van a enumerar las columnas que son meramente informativas y describir las columnas para las cuáles se ha tenido que aplicar un poco de lógica :

- **Continent:** Los valores que toma esta columna son todos los continentes pues los 100 diferentes países de los que hay datos pertenecen a cada uno de los continentes. Este dato no se usa para nada, es simplemente informativo.
- **GeographicalGroup\_UNU:** Los valores que toma esta columna son cada uno de los grupos geográficos. Hay un total de 25 grupos. Este dato es simplemente informativo pero se ha usado al hacer el preproceso manual para la red binaria.
- **Economic group:** Esta columna describe básicamente si el país pertenece a un grupo económico. En el caso de que pertenezca, tendrá el valor de "BRICS" o de "Triadization" y en el caso de que no pertenezca tendrá un "N". A modo de resumen, "Triadization" es aplicable a los países que han sufrido una globalización[14]. Sin embargo, "BRICS" es la sigla de Brasil, Rusia, India, China y Sudáfrica. Se consideran las 5 economías emergentes[13]. Es por eso, que en la red binaria se le ha dado un valor de 1 a "BRICS" y "Triadization" y un valor de -1 a "N". Sin embargo, al tener un tercer estado en la red ternaria, podemos hacer esa distinción de las economías emergentes, por lo tanto, "Triadization" recibe un 1, "BRICS" un 0 y "N" un -1.
- **DefaultMaritalPropertyRegime:** Esta columna describe lo que pertenece a cada uno de los esposos durante el matrimonio y cómo los activos y deudas se dividen en caso de divorcio o muerte[15]. En este caso esta columna puede tomar hasta 5 valores. Para la red binaria toma el valor de 1 cuando esta columna tiene el valor de " (a) ", " (b) " o " (c) " y -1 para los valores " (d) " y " (e) ". En la red ternaria tenemos el valor intermedio que se asigna al intermedio de los 5, es decir, un 1 para " (a) " y " (b) ", un 0 para " (c) " y un -1 para " (d) " y " (e) ".

## **C ¿Qué son los PPS?**

El estándar de poder adquisitivo (purchasing power standard) is una divisa ficticia creada por el instituto estadístico comunitario (Eurostat) cuyo valor se actualiza cada año y con el que, teóricamente, se puede comprar la misma cantidad de bienes y servicios en cada país. [16]